ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Master Project

FETI-DP Domain Decomposition Method

Author:
Christoph Jäggli

Supervisors:
Davide Forti
Dr. Simone Deparis
Prof. Alfio Quarteroni

A thesis submitted in fulfilment of the requirements for the degree of Master in Mathematical Engineering

in the

Chair of Modelling and Scientific Computing
Mathematics Section

June 2014

Declaration of Authorship

I, Christoph JÄGGLI, declare that this thesis titled, 'FETI-DP Domain Decomposition Method' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Abstract

School of Basic Science
Mathematics Section

Master in Mathematical Engineering

FETI-DP Domain Decomposition Method

by Christoph JÄGGLI

FETI-DP is a dual iterative, nonoverlapping domain decomposition method. By a Schur complement procedure, the solution of a boundary value problem is reduced to solving a symmetric and positive definite dual problem in which the variables are directly related to the continuity of the solution across the interface between the subdomains. The dual problem is solved by the conjugate gradient method with a Dirichlet preconditioner. In each iteration step a relatively small number of continuity constraints are enforced by the solution of a coarse level problem. Some numerical experiments are provided for the FETI-DP method applied to a 2- and 3-dimensional Poisson problem, as well as to a 3-dimensional Navier-Stokes problem.

Acknowledgements

I would like to thank Davide Forti, Dr. Simone Deparis and Prof. Alfio Quarteroni for giving me the opportunity to carry out my master thesis in the CMCS group under their supervision. I specially would like to highlight the effort of Davide in supporting me with precious discussions, ideas and very useful hints.

A very special acknowledgement to my dear girlfriend Barbara for her huge patience and thoughtfulness, to my parents, my brothers and sisters and their partners, without whom it would not have been possible to study at EPFL.

Finally I am very grateful to the good friends that I met on the other side of the "Röschtigrabe" and who allowed to experience many beautiful moments in Lausanne.

Contents

Chapter 1

Introduction

Domain decomposition [10, 11, 15] generally refers to a strategy used in numerical analysis in which a partial differential equation is split into coupled problems on smaller subdomains. This splitting can be motivated at the continuous level by different physical models applying in different regions of the computational domain, or at the discretized level, to lower the computational complexity and parallelize the workload. A number of domain decomposition methods have been investigated mathematically and experimentally by applying them to a wide range of partial differential equations [4, 10–12, 15].

The main idea of domain decomposition is to subdivide the computational domain into a number of subdomains, in which discretized problems of smaller sizes are solved using an efficient parallel solution algorithm. The use of basic iterative methods for solving large linear systems often results in a very slow convergence rate. Instead of solving one huge problem it may be more convenient to solve simultaneously many smaller problems several times. Domain decomposition also allows to devise parallel preconditioner. A crucial point of such an algorithm is how the smaller subproblems communicate to each other. On one hand, this communication is tended to be kept as small as possible, such that the subproblems are mostly independently solvable in parallel. On the other hand, low communication negatively affects the convergence rate of the method. A relatively recent method, who successfully balances the requirement of a minimum of communication by keeping a fast convergence rate is the *Finite Element Tearing and Interconnecting Dual-Primal (FETI-DP)* method [1, 5, 7, 10, 13, 15]. FETI-DP is a dual iterative method using a set of non-overlapping substructures. The original problem is translated into a dual problem in which the local iterates are discontinuous across the biggest part of the interface between the subdomains and only a small number of continuity constraints are imposed. The iteration solves for a set of Lagrange multipliers ([8]) being dual variables aiming to enforce continuity across the entire interface.

The theoretical analysis for the FETI-DP method has been developed for the elliptic and autoadjoint Poisson problem (e.g. [10, 15] and more recently for linear elasticity ([1, 13]) and incompressible Stokes ([7]) problems. It can be shown that the FETI-DP method applied to those model problems is scalable with respect to the number of degrees of freedom (dofs) per subdomains, i.e. the number of iterations for solving the dual problem is independent of the number of subdomains, as long as the number of dofs per subdomain is not modified. One of the characteristics of the FETI-DP domain decomposition method is the relative high amount of preprocessing work that has to be done in order to identify vertices, edges and faces of the subdomains.

In this work we will first derive the structure of the FETI-DP method applied to the 2- and 3-dimensional elliptic and autoadjoint Poisson problem. We intent to doing so by using intuitive notations enriched with meaningful examples (Chapter 2). To conclude this chapter, some numerical results are provided, in which the scalability of the method can be observed. In Chapter 3 we adapt the FETI-DP method to the time dependent Navier-Stokes problem for an incompressible fluid. The method is implemented in Matlab using linear and continuous FE basis functions for the pressure and the velocity field and a Brezzi-Pitkäranta stabilization term.

All the work has been carried out in the Chair of Modelling and Scientific Computing (CMCS) at the École Polytechnique Fédérale de Lausanne in Switzerland.

Chapter 2

FETI-DP applied to Poisson's Equation

2.1 Problem Setting and Geometry

The presentation of the FETI-DP domain decomposition method in this chapter is based on the work of Toselli and Widlund [15], Quarteroni [10], Farhat, Lesoinne, LeTallec, Pierson and Rixen [1], Rheinbach [13] and Rixen and Farhat [14]. In this chapter we will firstly introduce the general structure of the method applied to an elliptic, second order and autoadjoint problem, and then we will apply it to 2- and 3-dimensional problems. We denote by $\Omega \subset \mathbb{R}^d$ the computational domain (with $d = 2, 3$) such that $|\Omega| < +\infty$. We denote the boundary of Ω by $\partial\Omega$ and split it into the subsets Γ_D and Γ_N, where Dirichlet and Neumann conditions apply, respectively, such that $\overset{\circ}{\Gamma}_D \cap \overset{\circ}{\Gamma}_N = \emptyset$ and $\partial\Omega = \overline{\Gamma_D \cup \Gamma_N}$; we also indicate with $\hat{\boldsymbol{n}}$ the outward unit vector normal to $\partial\Omega$. For sufficiently regular functions $f : \Omega \to \mathbb{R}$, $g : \Gamma_D \to \mathbb{R}$ and $h : \Gamma_N \to \mathbb{R}$ the *Poisson problem* reads:

$$\text{find } u : \Omega \to \mathbb{R} \quad : \quad \begin{cases} -\nabla \cdot (\rho \nabla u) = f & \text{in } \Omega, \\ u = g & \text{on } \Gamma_D, \\ \dfrac{\partial u}{\partial \hat{\boldsymbol{n}}} = h & \text{on } \Gamma_N. \end{cases} \tag{2.1}$$

We assume that $\rho(x) \geq \rho_0 > 0$.

Let us introduce the function space $L^2(\Omega)$, i.e. the space of square integrable functions in Ω, and the Hilbert space $H^1(\Omega) = \{v : \Omega \to \mathbb{R},\ D^\alpha v \in L^2(\Omega) \text{ and } \forall \alpha \in \mathbb{N}^d, |\alpha| \leq 1\}$, where $\alpha = \{\alpha_1, \ldots, \alpha_d\} \in \mathbb{N}^d$ is a multi-index, $|\alpha| = \sum_{i=1}^{d} \alpha_i$ and D^α the multi-index

distributional derivative operator; see e.g. [6]. We set the test function space $W := \{w \in H^1(\Omega),\ w|_{\Gamma_D} = 0\}$. Supposing that $G : \overline{\Omega} \to \mathbb{R}$ is a sufficiently regular lifting function such that $G|_{\Gamma_D} = g$, the weak form of the Eq. (2.1) ([10, 11]) reads:

$$\text{find } u \in W \quad : \quad a(u, w) = F(w) - a(G, w) \qquad \forall w \in W \tag{2.2}$$

where $a : W \times W \to \mathbb{R}$ is the bilinear form:

$$a(u, w) = \int_\Omega \rho \nabla u \cdot \nabla w d\Omega \tag{2.3}$$

and $F : W \to \mathbb{R}$ is the linear functional:

$$F(w) = \int_\Omega f w d\Omega + \int_{\Gamma_N} h w d\Gamma_N \tag{2.4}$$

In the framework of the FETI-DP method, the computational domain $\Omega \subset \mathbb{R}^d$ is split into a set of non overlapping and open subdomains, denoted by $\Omega^{(s)}$, $s = 1, \ldots, N_s$, being N_s the number of subdomains. The pairwise intersections of the subdomains are empty, although the union of the closure of the subdomains has to be equal to the closure of the computational domain, i.e.:

$$\Omega^{(s)} \cap \Omega^{(t)} = \emptyset, \quad s \neq t$$
$$\bigcup_{s=1}^{N_s} \overline{\Omega}^{(s)} = \overline{\Omega}.$$

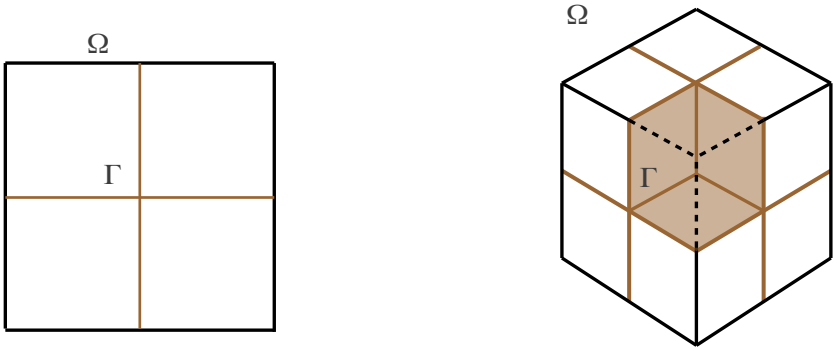
The partitioning of the computational domain into a set of N_s subdomains induces an interface Γ being the interconnecting layer of common boundary parts, defined by:

$$\Gamma := \bigcup_{s \neq t} \partial\Omega^{(s)} \cap \partial\Omega^{(t)}.$$

For an illustration of the interface in a 2- and 3-dimensional domain see Figure 2.1. The interface Γ is the union of

- faces, regarded as open sets and being shared by two subdomains (only in 3-D),
- edges, regarded as open sets and being shared by three or more subdomains,
- vertices, being endpoints of edges.

If there is an edge which is part of Γ_N and which is shared by two subdomains, we will regard that edge as a part of the face common to this pair of subdomains. When solving numerically a boundary value problem for a partial differential equation (PDE) in the



(A) The interface in a 2-dimensional square decomposed into 4 subdomains.

(B) The interface in a 3-dimensional cube decomposed into 8 subdomains.

FIGURE 2.1: Illustration of the interface in a 2- and 3-dimensional domain.

framework of the Galerkin method [9–11], a polygonal mesh is generated in order to represent the geometric domain Ω. For $s = 1, \ldots, N_s$ let $\tau_h^{(s)}$ be a quasi uniform mesh of the subdomain $\Omega^{(s)}$ and we consider conforming discretizations at the subdomain interfaces. Let $W^h(\Omega^{(s)}) = X_h^r$ be the standard finite element (FE) space of continuous, piecewise polynomial basis functions of degree r [9–11] in the subdomain $\Omega^{(s)}$. According to the weak formulation of the problem (2.2) we consider only basis functions vanishing on the Dirichlet boundary Γ_D. Let the dimension of the space $W^h(\Omega^{(s)})$ be given by $n^{(s)}$. If we extend the local basis functions defined in $W^h(\Omega^{(s)})$ by zero on $\overline{\Omega}\backslash\overline{\Omega}^{(s)}$ we can define an unusual FE-space $W^h(\Omega)$ on Ω, that is the cartesian product of the spaces defined on the local subdomains:

$$W^h(\Omega) := \prod_{s=1}^{N_s} W^h(\Omega^{(s)}).$$

The dimension of the global FE-space $W^h(\Omega)$ is then given by $N_W = \sum_{s=1}^{N_s} n^{(s)}$. For the sake of simplicity, we will not distinguish between the two notations $u \in W^h(\Omega)$ and $\boldsymbol{u} \in \mathbb{R}^{N_W}$ and we denote by u_i the coefficient associated to the i-th basis function. Moreover, we highlight that $\boldsymbol{u} \in W^h(\Omega)$ is not necessary continuous across the interface Γ. The nodes induced by the triangulation $\tau_h := \bigcup_{s=1}^{N_s} \tau^{(s)}$ which lie on the interface Γ are split in two groups: the *dual* nodes (henceforth denoted by the subscript Δ) and the *primal* nodes (subscript Π). The space $\tilde{W}^h(\Omega)$ is defined as the subspace of $W^h(\Omega)$ formed by the continuous basis functions across all the primal nodes. For a given node x we denote by $\mathcal{N}_x \subset \{1, \ldots, N_s\}$ the set of indices of all subdomains to which x belongs

to. If $\mathcal{P} = \{x_{\Pi,i},\, i = 1, \ldots, N_\Pi\}$ is the set of primal nodes, the space $\tilde{W}^h(\Omega)$ reads:

$$\tilde{W}^h(\Omega) = \{\boldsymbol{v} \in W^h(\Omega) \;:\; \boldsymbol{v}^{(s)}(x) = \boldsymbol{v}^{(t)}(x),\, \forall s,t \in \mathcal{N}_x,\, \forall x \in \mathcal{P}\} \tag{2.5}$$

We denote by the subscript I the set of internal nodes which are not on the interface. A node lying on the intersection $\Gamma_N \cap \partial\Omega^{(s)}$ for a subdomain $\Omega^{(s)}$ is either lying on the interface Γ (and therefore a primal or a dual node) or added to the internal nodes of $\Omega^{(s)}$. Further, we use the subscript "R" as the set of nodes which are not primal nodes (i.e. the set of remaining nodes). The dimension of the space $\tilde{W}^h(\Omega)$ is $N_{\tilde{W}} := N_R + N_\Pi$ where $N_R = \sum_{s=1}^{N_s} n_R^{(s)}$ and $n_R^{(s)}$ is the number of remaining nodes in the subdomain $\overline{\Omega}^{(s)}$.

Let $A \in \mathbb{R}^{N_W \times N_W}$ and $\boldsymbol{f} \in \mathbb{R}^{N_W}$ be the stiffness matrix and the vector of applied loads, assembled by means of the Galerkin method in the space $W^h(\Omega)$, i.e.:

$$A = \begin{pmatrix} A^{(1)} & & \\ & \ddots & \\ & & A^{(N_s)} \end{pmatrix} \quad \text{and} \quad \boldsymbol{f} = \begin{pmatrix} \boldsymbol{f}^{(1)} \\ \vdots \\ \boldsymbol{f}^{(N_s)} \end{pmatrix}.$$

Let $P_{\tilde{W}} \in \{0,1\}^{N_W \times N_{\tilde{W}}}$ be a boolean coupling operator such that $P_{\tilde{W}}$ defines a one-to-one map between $\mathbb{R}^{N_{\tilde{W}}}$ and $\tilde{W}^h(\Omega)$. The stiffness matrix $\tilde{A} \in \mathbb{R}^{N_{\tilde{W}} \times N_{\tilde{W}}}$ and the vector of applied loads $\tilde{\boldsymbol{f}} \in \mathbb{R}^{N_{\tilde{W}}}$ assembled in $\tilde{W}^h(\Omega)$ are obtained by applying the coupling operator $P_{\tilde{W}}$ to A and to $\boldsymbol{f}$ such that $\tilde{A} = P_{\tilde{W}}^{\mathrm{T}} A P_{\tilde{W}}$ and $\tilde{\boldsymbol{f}} = P_{\tilde{W}}^{\mathrm{T}} \boldsymbol{f}$. The nodes can then be reordered such that:

$$\tilde{A} = \begin{pmatrix} A_{RR} & \tilde{A}_{R\Pi} \\ \tilde{A}_{R\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} \boldsymbol{u}_R \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} \quad \text{and} \quad \boldsymbol{f} = \begin{pmatrix} \boldsymbol{f}_R \\ \tilde{\boldsymbol{f}}_\Pi \end{pmatrix}$$

with

$$A_{RR} = \begin{pmatrix} A_{RR}^{(1)} & & \\ & \ddots & \\ & & A_{RR}^{(N_s)} \end{pmatrix}, \quad \boldsymbol{u}_R = \begin{pmatrix} \boldsymbol{u}_R^{(1)} \\ \vdots \\ \boldsymbol{u}_R^{(N_s)} \end{pmatrix} \quad \text{and} \quad \boldsymbol{f}_R = \begin{pmatrix} \boldsymbol{f}_R^{(1)} \\ \vdots \\ \boldsymbol{f}_R^{(N_s)} \end{pmatrix}.$$

Remark 2.1. Let $G = (V, E)$ be an undirected graph where the vertices $v \in V$ are the subdomains of Ω: an edge $e = (s,t) \in E$ exists if and only if there exists a primal node common to $\tau_h^{(s)}$ and $\tau_h^{(t)}$. From an algebraic point of view, if the graph G is connected, then the local sub-matrices $A_R^{(s)}$ are non-singular (c.f. [13]). In this chapter we will only consider problems with non-singular matrices $A_R^{(s)}$.

Remark 2.2. Another way to obtain the matrices $\tilde{A}_{R\Pi}$ and $\tilde{A}_{\Pi\Pi}$, is to assemble the local stiffness matrices

$$A^{(s)} = \begin{pmatrix} A_{RR}^{(s)} & A_{R\Pi}^{(s)} \\ {A_{R\Pi}^{(s)}}^{\mathrm{T}} & A_{\Pi\Pi}^{(s)} \end{pmatrix}$$

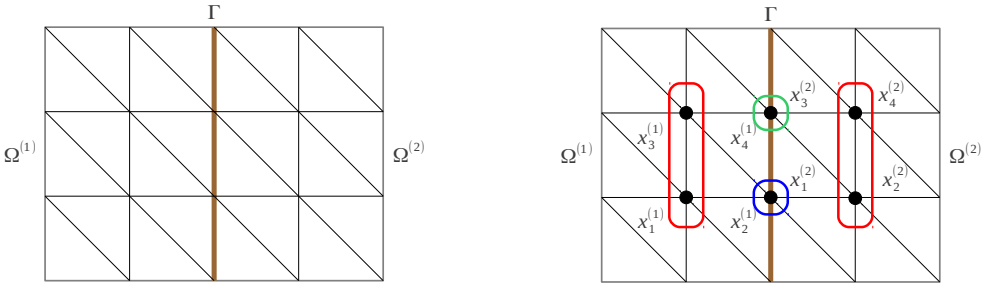
in the local spaces $W^h(\Omega^{(s)})$ and to define, for each subdomain $s = 1, \dots, N_s$, two boolean operators $P_R^{(s)} \in \{0,1\}^{n_R^{(s)} \times N_R}$ and $P_\Pi^{(s)} \in \{0,1\}^{n_\Pi^{(s)} \times N_\Pi}$ such that:

$$P_R^{(s)} \boldsymbol{u}_R = \boldsymbol{u}_R^{(s)} \quad \text{and} \quad P_\Pi^{(s)} \tilde{\boldsymbol{u}}_\Pi = \boldsymbol{u}_\Pi^{(s)}$$

with $n_\Pi^{(s)}$ being the number of primal nodes in $\tau_h^{(s)}$. By construction of $\tilde{W}^h(\Omega^{(s)})$, it is easy to see that:

$$\tilde{A}_{R\Pi} = \sum_{s=1}^{N_s} {P_R^{(s)}}^{\mathrm{T}} A_{R\Pi}^{(s)} P_\Pi^{(s)} \quad \text{and} \quad \tilde{A}_{\Pi\Pi} = \sum_{s=1}^{N_s} {P_\Pi^{(s)}}^{\mathrm{T}} A_{\Pi\Pi}^{(s)} P_\Pi^{(s)}$$

Example 2.1. *To get a better understanding of the definitions given above, we introduce an easy example. Let the computational domain $\Omega \subset \mathbb{R}^2$ be a rectangle with $\partial\Omega = \Gamma_D$, partitioned in two subdomains $\Omega^{(1)}$ and $\Omega^{(2)}$. Both subdomains are discretized by a structured triangular mesh conforming at the interface $\Gamma = \overline{\Omega}^{(1)} \cap \overline{\Omega}^{(2)}$ as illustrated in Figure 2.2a. As $\partial\Omega = \Gamma_D$, all the dofs associated to the nodes on the boundary are eliminated. The two nodes lying on the interface Γ are separated into a primal and a dual one, while in the interior of each subdomain there are two internal nodes (see Figure 2.2b). In this case, for $s = 1, 2$, we can see that $\boldsymbol{u}^{(s)} \in W^h(\Omega^{(s)})$ is defined by*



(A) The two subdomains $\Omega^{(1)}$ and $\Omega^{(2)}$ are partitioned into structured triangular meshes matching across the interface Γ.

(B) The nodes are grouped into primal nodes Π (green), dual nodes Δ (blue) and internal nodes I (red).

FIGURE 2.2: Example of a two-dimensional computational domain partitioned into two subdomains.

$$\boldsymbol{u}^{(s)} = \begin{pmatrix} u_1^{(s)} \\ \vdots \\ u_4^{(s)} \end{pmatrix}$$

and therefore, $\boldsymbol{u} \in W^h(\Omega) = W^h(\Omega^{(1)}) \times W^h(\Omega^{(2)})$ *reads*

$$\boldsymbol{u} = (u_1^{(1)}, u_2^{(1)}, u_3^{(1)}, u_4^{(1)}, u_1^{(2)}, u_2^{(2)}, u_3^{(2)}, u_4^{(2)})^{\mathrm{T}}$$

The dimension of the spaces are given by $n^{(1)} = n^{(2)} = 4$, $n_R^{(1)} = n_R^{(2)} = 3$, $N_R = n_R^{(1)} + n_R^{(2)} = 6$ *and* $N_\Pi = 1$, *such that*

$$N_W = n^{(1)} + n^{(2)} = 8, \quad \textit{and} \quad N_{\tilde{W}} = N_R + N_\Pi = 7.$$

By definition, $\tilde{\boldsymbol{u}} \in \tilde{W}^h(\Omega)$ *is continuous across each primal node, i.e.*

$$\tilde{\boldsymbol{u}} = (u_1^{(1)}, u_2^{(1)}, u_3^{(1)}, u_\Pi, u_1^{(2)}, u_2^{(2)}, u_\Pi, u_4^{(2)})^{\mathrm{T}}.$$

The coupling operator $P_{\tilde{W}} \in \{0,1\}^{8\times 7}$ *is such that*

$$P_{\tilde{W}} = \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & 1 & & & \\ & & & & 1 & & \\ & & & & & 1 & \\ & & & 1 & & & \\ & & & & & & 1 \end{pmatrix}.$$

We need to enforce a set of continuity constraints to guarantee the continuity of $\boldsymbol{u} \in \tilde{W}^h(\Omega)$ at the dual nodes. The total number of continuity constraints, n_{cc}, highly depends on the choice of the dual nodes. If a node x belongs to $m > 1$ subdomains, we can use between $m-1$ and $\dfrac{m(m-1)}{2}$ continuity conditions in order to guarantee the continuity of the solution at the node x (see Figure 2.3). The continuity conditions are expressed through a boolean matrix B_R:

$$B_R \in \{0, -1, 1\}^{n_{cc}\times N_R}$$

such that the values of $\tilde{\boldsymbol{u}} = (\boldsymbol{u}_R^{\mathrm{T}}, \tilde{\boldsymbol{u}}_\Pi^{\mathrm{T}})^{\mathrm{T}}$ associated to the dual nodes coincide, if and only if

$$B_R \boldsymbol{u}_R = 0.$$

Example 2.2. *Let us consider a dual node* x *belonging to 4 subdomains as illustrated in Figure 2.3. The vector of "remaining" dofs in the solution* $\tilde{\boldsymbol{u}} = (\boldsymbol{u}_R^{\mathrm{T}}, \tilde{\boldsymbol{u}}_\Pi^{\mathrm{T}})^{\mathrm{T}}$ *is*

$$\boldsymbol{u}_R = (\cdots, u_4^{(1)}, u_2^{(2)}, u_1^{(3)}, u_3^{(4)}, \cdots)^{\mathrm{T}},$$

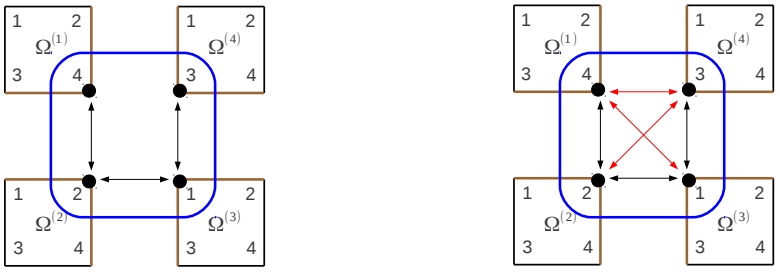


FIGURE 2.3: The number of continuity conditions associated to a dual node corresponding to subdomains in the nonredundant (left) and the fully redundant (right) case

and therefore a non-redundant (nr) and a fully redundant (fr) continuity operator are given by

$$B_R^{nr} = \begin{pmatrix} 1 & -1 & 0 & 0 & \\ \cdots 0 & 1 & -1 & 0 & \cdots \\ 0 & 0 & 1 & -1 & \end{pmatrix} \quad \textit{and} \quad B_R^{fr} = \begin{pmatrix} 1 & -1 & 0 & 0 & \\ 1 & 0 & -1 & 0 & \\ \cdots 1 & 0 & 0 & -1 & \cdots \\ 0 & 1 & -1 & 0 & \\ 0 & 1 & 0 & -1 & \\ 0 & 0 & 1 & -1 & \end{pmatrix}$$

Using the equivalence described in Appendix A, we can write the discretized problem in form of the saddle point system:

$$\begin{pmatrix} A_{RR} & \tilde{A}_{R\Pi} & B_R^{\mathrm{T}} \\ \tilde{A}_{R\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} & 0 \\ B_R & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_R \\ \tilde{\boldsymbol{u}}_\Pi \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_R \\ \tilde{\boldsymbol{f}}_\Pi \\ 0 \end{pmatrix}. \tag{2.6}$$

After eliminating the values corresponding to the remaining (R) and to the primal (Π) dofs, we obtain the *dual problem*:

$$\begin{cases} \boldsymbol{u}_R = A_{RR}^{-1}\left(\boldsymbol{f}_R - \tilde{A}_{R\Pi}\boldsymbol{u}_\Pi - B_R^{\mathrm{T}}\boldsymbol{\lambda}\right) \\ \tilde{\boldsymbol{u}}_\Pi = \tilde{S}_\Pi^{-1}\left(\tilde{\boldsymbol{f}}_\Pi - \tilde{A}_{R\Pi}^{\mathrm{T}}A_{RR}^{-1}\boldsymbol{f}_R + \tilde{A}_{R\Pi}^{\mathrm{T}}A_{RR}^{-1}B_R^{\mathrm{T}}\boldsymbol{\lambda}\right) \\ F\boldsymbol{\lambda} = \boldsymbol{d} \end{cases} \tag{2.7}$$

where

$$\begin{aligned} F &= F_{\lambda_{RR}} + F_{\lambda_{R\Pi}} \tilde{S}_\Pi^{-1} F_{\lambda_{R\Pi}}^{\mathrm{T}}, \\ \boldsymbol{d} &= B_R A_{RR}^{-1} \boldsymbol{f}_R + F_{\lambda_{R\Pi}} \tilde{S}_\Pi^{-1} \left(\tilde{A}_{R\Pi}^{\mathrm{T}} A_{RR}^{-1} \boldsymbol{f}_R - \boldsymbol{f}_\Pi \right), \\ \tilde{S}_\Pi &= \tilde{A}_{\Pi\Pi} - \tilde{A}_{R\Pi}^{\mathrm{T}} A_{RR}^{-1} \tilde{A}_{R\Pi}, \\ F_{\lambda_{RR}} &= B_R A_{RR}^{-1} B_R^{\mathrm{T}}, \\ F_{\lambda_{R\Pi}} &= B_R A_{RR}^{-1} \tilde{A}_{R\Pi}. \end{aligned} \tag{2.8}$$

Remark 2.3. Thanks to Remark 2.1 we assume A_{RR} to be non-singular. The matrix $\tilde{S}_\Pi$ in (2.8) is the *Schur complement* of A_{RR} in A. In order to show that $\tilde{S}_\Pi$ is non-singular, one can use the following result given in [2]:

Proposition 2.4. *Let $M \in \mathbb{R}^{n\times n}$ be a symmetric matrix of the form:*

$$M = \begin{pmatrix} A & B \\ B^{\mathrm{T}} & C \end{pmatrix}$$

where $m \in \mathbb{N}$, $A \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times(n-m)}$ and $C \in \mathbb{R}^{(n-m)\times(n-m)}$; if A is invertible then:

1. *M is positive definite if and only if A and $C - B^{\mathrm{T}}A^{-1}B$ are positive definite.*
2. *If A is positive definite then M is positive semidefinite if and only if $C - B^{\mathrm{T}}A^{-1}B$ is positive semidefinite.*

2.2 Dirichlet Preconditioner for the Dual Problem

In this Section we recall the Dirichlet preconditioner for the dual problem (2.7). It has been introduced in [10, 13, 15] and we will show in the next section that it is mathematically optimal in the sense that the preconditioned dual problem is scalable with respect to $\frac{H}{h}$ where H and h are the characteristic subdomain and mesh sizes, respectively.

The diffusion function $\rho = \rho(x)$ in Eq. (2.1) is assumed to be constant within each subdomain, i.e. for $s = 1, \ldots, N_s$, $\rho|_{\Omega^{(s)}} = \rho_s$. An important role in the description of the preconditioner is played by a set of weighting functions $\delta_s^\dagger$ defined on the interface Γ. As above, for a node $x \in \Gamma \cap \overline{\Omega}^{(s)}$, let $\mathcal{N}_x$ denote the set of indices of all subdomains to which x belongs to. We define the functions

$$\delta_s^\dagger(x) := \frac{\rho_s}{\sum_{t\in\mathcal{N}_x} \rho_t}. \tag{2.9}$$

In the case of ρ being constant all over the computational domain Ω, the functions $\delta_s^\dagger(x)$ becomes

$$\delta_s^\dagger(x) = \frac{1}{|\mathcal{N}_x|}.$$

We can see immediately that for a given x the functions $\delta_s^\dagger$ satisfy the partition of unity, i.e.

$$\sum_{s\in\mathcal{N}_x} \delta_s^\dagger(x) = 1.$$

Let $D^{(s)}$, for $s = 1,\dots,N_s$, be a diagonal scaling matrix of size $n_{cc} \times n_{cc}$, whose k-th diagonal entry is $\delta_t^\dagger(x)$ if and only if the k-th row in B_R enforces continuity of the two solutions $\boldsymbol{u}^{(s)}$ and $\boldsymbol{u}^{(t)}$ across the node x. We can now define a scaled continuity operator $B_{R,D} \in \mathbb{R}^{n_{cc}\times N_R}$ such that:

$$B_{R,D} = \left(D^{(1)}B_R^{(1)},\dots,D^{(N_s)}B_R^{(N_s)}\right) \tag{2.10}$$

where $B_R^{(s)} \in \{0,1,-1\}^{n_{cc}\times n_R^{(s)}}$ denotes the submatrix of B_R corresponding to the subdomain $\Omega^{(s)}$. As the remaining (R) dofs are the union of the dual (Δ) and the internal (I) ones we can see the operator $B_R^{(s)}$ as an extension by zeros of the operator $B_\Delta^{(s)} \in \{0,1,-1\}^{n_{cc}\times n_\Delta^{(s)}}$ acting only on the dual part $\boldsymbol{u}_\Delta^{(s)}$ of the solution $\boldsymbol{u}^{(s)}$. $n_\Delta^{(s)}$ indicates the number of dual dofs corresponding to the subdomain $\Omega^{(s)}$. For $s = 1,\dots,N_s$ we denote by $S_{I\Delta,\Delta}^{(s)} \in \mathbb{R}^{n_\Delta^{(s)}\times n_\Delta^{(s)}}$ the local Schur complement

$$S_{I\Delta,\Delta}^{(s)} = A_{\Delta\Delta}^{(s)} + {A_{I\Delta}^{(s)}}^{\mathrm{T}} {A_{II}^{(s)}}^{-1} A_{I\Delta}^{(s)}.$$

Then, the *Dirichlet preconditioner* $M^{-1} \in \mathbb{R}^{n_{cc}\times n_{cc}}$ reads:

$$M^{-1} := \sum_{s=1}^{N_s} D^{(s)} B_\Delta^{(s)} S_{I\Delta,\Delta}^{(s)} {B_\Delta^{(s)}}^{\mathrm{T}} D^{(s)}. \tag{2.11}$$

We will show in Theorem 2.9 that the condition number of the preconditioned dual matrix can be bounded by a bound depending only on the ratio H/h where H and h are the characteristic subdomain and mesh sizes, respectively.

2.3 Solving the Preconditioned Dual Problem

The main idea of domain decomposition techniques is to solve a problem using an efficient parallel strategy. In this section we describe a way how the preconditioned dual problem (2.7) can be solved using a parallel implementation. First, by making use of the following propositions, we can show that the matrix F in Eq. (2.8) is symmetric and positive

definite. The Proposition 2.5 is taken from [2] while Proposition 2.6 is new and for the proof of proposition 2.7 we have been inspired by the proof of Lemma 4 in [7].

Proposition 2.5. *Let us consider a symmetric 2 by 2 block matrix as in Proposition 2.4. If A, C and both Schur complements $S_{AC} = A - BC^{-1}B^{\mathrm{T}}$ and $S_{CA} = C - B^{\mathrm{T}}A^{-1}B$ are invertible, we have:*

$$(A - BC^{-1}B^{\mathrm{T}})^{-1} = A^{-1} + A^{-1}B(C - B^{\mathrm{T}}A^{-1}B)^{-1}B^{\mathrm{T}}A^{-1},$$
$$(C - B^{\mathrm{T}}A^{-1}B)^{-1} = C^{-1} + C^{-1}B^{\mathrm{T}}(A - BC^{-1}B^{\mathrm{T}})^{-1}BC^{-1}.$$

Proposition 2.6. *The Schur complement of $A_{\Delta\Delta}$ in*

$$\tilde{A} = \begin{pmatrix} A_{II} & A_{I\Delta} & \tilde{A}_{I\Pi} \\ A_{I\Delta}^{\mathrm{T}} & A_{\Delta\Delta} & \tilde{A}_{\Delta\Pi} \\ \tilde{A}_{I\Pi}^{\mathrm{T}} & \tilde{A}_{\Delta\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} \end{pmatrix}$$

is obtained by:

$$\tilde{S}_{\Delta} = A_{\Delta\Delta} - \begin{pmatrix} A_{I\Delta}^{\mathrm{T}} & \tilde{A}_{\Delta\Pi} \end{pmatrix} \begin{pmatrix} A_{II} & \tilde{A}_{I\Pi} \\ \tilde{A}_{I\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} \end{pmatrix}^{-1} \begin{pmatrix} A_{I\Delta} \\ \tilde{A}_{\Delta\Pi}^{\mathrm{T}} \end{pmatrix}.$$

If F is the matrix of the dual problem defined in Eq. (2.8) and, as above, B_{Δ} is the restriction of the continuity operator B_R to the set of dual degrees of freedom (Δ) we have:

$$F = B_{\Delta}\tilde{S}_{\Delta}^{-1}B_{\Delta}^{\mathrm{T}}.$$

Proof. We will show that for all $\boldsymbol{\lambda}, \boldsymbol{\mu} \in \mathbb{R}^{n_{cc}}$:

$$\boldsymbol{\mu}^{\mathrm{T}}F\boldsymbol{\lambda} = \boldsymbol{\mu}^{\mathrm{T}}B_{\Delta}\tilde{S}_{\Delta}^{-1}B_{\Delta}^{\mathrm{T}}\boldsymbol{\lambda}$$

and therefore $F = B_{\Delta}\tilde{S}_{\Delta}^{-1}B_{\Delta}^{\mathrm{T}}$. By construction of the continuity operator:

$$B_R^{\mathrm{T}}\boldsymbol{\lambda} = \begin{pmatrix} 0 \\ B_{\Delta}^{\mathrm{T}}\boldsymbol{\lambda} \end{pmatrix}$$

and therefore, for all $\boldsymbol{u}_R = \begin{pmatrix} \boldsymbol{u}_I^{\mathrm{T}} & \boldsymbol{u}_{\Delta}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^{N_R}$:

$$\boldsymbol{\lambda}^{\mathrm{T}}B_R\boldsymbol{u}_R = \begin{pmatrix} 0 & \boldsymbol{\lambda}^{\mathrm{T}}B_{\Delta} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_{\Delta} \end{pmatrix} = \boldsymbol{\lambda}^{\mathrm{T}}B_{\Delta}\boldsymbol{u}_{\Delta}.$$

By the definition of F in Eq. (2.8) and by the Proposition 2.5 it is easy to see that

$$F = B_R \left(A_{RR}^{-1} + A_{RR}^{-1} \tilde{A}_{R\Pi} S_{\Pi\Pi}^{-1} \tilde{A}_{R\Pi}^{\mathrm{T}} A_{RR}^{-1} \right) B_R^{\mathrm{T}} = B_R \tilde{S}_R^{-1} B_R^{\mathrm{T}},$$

where $\tilde{S}_R = A_{RR} - \tilde{A}_{R\Pi} A_{\Pi\Pi}^{-1} \tilde{A}_{R\Pi}^{\mathrm{T}}$. Let us remark that if $M \in \mathbb{R}^{n\times n}$ is a symmetric matrix of the form

$$M = \begin{pmatrix} A & B \\ B^{\mathrm{T}} & C \end{pmatrix},$$

where $m \in \mathbb{N}$, $A \in \mathbb{R}^{m\times m}$ is non singular, $B \in \mathbb{R}^{m\times(n-m)}$ and $C \in \mathbb{R}^{(n-m)\times(n-m)}$, then, applying the inverse of the Schur complement $S_{CA} = C - B^{\mathrm{T}} A^{-1} B$ to an arbitrary vector $\boldsymbol{y} \in \mathbb{R}^{n-m}$ is equivalent to solve a system with the original matrix M and suitable right hand side, i.e.

$$\boldsymbol{x} = S_{CA}^{-1} \boldsymbol{y} \qquad \Longleftrightarrow \qquad \begin{pmatrix} A & B \\ B^{\mathrm{T}} & C \end{pmatrix} \begin{pmatrix} \boldsymbol{z} \\ \boldsymbol{x} \end{pmatrix} = \begin{pmatrix} 0 \\ \boldsymbol{y} \end{pmatrix}$$

The proof is concluded by observing that:

$$\begin{aligned} \text{compute} \quad \boldsymbol{\mu}^{\mathrm{T}} F \boldsymbol{\lambda} = \boldsymbol{\mu}^{\mathrm{T}} B_R \tilde{S}_R^{-1} B_R^{\mathrm{T}} \boldsymbol{\lambda} \quad &\Longleftrightarrow \quad \begin{cases} \text{solve} \quad \tilde{A} \begin{pmatrix} \boldsymbol{u}_R \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} = \begin{pmatrix} B_R^{\mathrm{T}} \boldsymbol{\lambda} \\ 0 \end{pmatrix} \\ \text{and compute} \quad \boldsymbol{\mu}^{\mathrm{T}} B_R \boldsymbol{u}_R \end{cases} \\ &\Longleftrightarrow \quad \begin{cases} \text{solve} \quad \tilde{A} \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_\Delta \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} = \begin{pmatrix} 0 \\ B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \\ 0 \end{pmatrix} \\ \text{and compute} \quad \boldsymbol{\mu}^{\mathrm{T}} B_\Delta \boldsymbol{u}_\Delta \end{cases} \\ &\Longleftrightarrow \quad \text{compute} \quad \boldsymbol{\mu}^{\mathrm{T}} B_\Delta \tilde{S}_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \end{aligned}$$

□

Now we are ready to proof the following proposition.

Proposition 2.7. *The matrix F in Eq. (2.8) is symmetric and positive definite in the space* $\mathrm{Im}(B_\Delta)$.

Proof. The matrix $\tilde{A}$ is symmetric and positive definite, as it is a discrete representation in $\tilde{W}^h(\Omega)$ of a direct sum of local Poisson problems in $\Omega^{(s)}$ with Dirichlet boundary condition on $\Gamma_D \cap \partial\Omega^{(s)}$ and on the primal part of $\Gamma \cap \partial\Omega^{(s)}$. The matrix

$$\begin{pmatrix} A_{II} & \tilde{A}_{I\Pi} \\ \tilde{A}_{I\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} \end{pmatrix},$$

being a principal submatrix of $\tilde{A}$, is symmetric and positive definite too. From this, it follows that for any $\boldsymbol{u}_\Delta$, there is a vector

$$\begin{pmatrix} \boldsymbol{u}_I \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} = - \begin{pmatrix} A_{II} & \tilde{A}_{I\Pi} \\ \tilde{A}_{I\Pi}^{\mathrm{T}} & \tilde{A}_{\Pi\Pi} \end{pmatrix}^{-1} \begin{pmatrix} A_{I\Delta} \\ \tilde{A}_{\Delta\Pi}^{\mathrm{T}} \end{pmatrix} \boldsymbol{u}_\Delta$$

such that

$$\tilde{A} \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_\Delta \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} = \begin{pmatrix} 0 \\ \tilde{S}_\Delta \boldsymbol{u}_\Delta \\ 0 \end{pmatrix}$$

and therefore, for $\boldsymbol{u}_\Delta \neq 0$

$$\boldsymbol{u}_\Delta^{\mathrm{T}} \tilde{S}_\Delta \boldsymbol{u}_\Delta = \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_\Delta \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} 0 \\ \tilde{S}_\Delta \boldsymbol{u}_\Delta \\ 0 \end{pmatrix} = \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_\Delta \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix}^{\mathrm{T}} \tilde{A} \begin{pmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_\Delta \\ \tilde{\boldsymbol{u}}_\Pi \end{pmatrix} > 0.$$

This shows that $\tilde{S}_\Delta$ and also $\tilde{S}_\Delta^{-1}$ are symmetric and positive definite. By the closed rank theorem, the space $\mathbb{R}^{n_{cc}}$ is such that:

$$\mathbb{R}^{n_{cc}} = \mathrm{Ker}(B_\Delta^{\mathrm{T}}) \oplus \mathrm{Im}(B_\Delta)$$

what concludes the proof. □

This last proposition allows us to use a *Preconditioned Conjugate Gradient (PCG)* method (see [10, 15]) for solving the dual problem (2.7). The initial guess $\boldsymbol{\lambda}_0$ needs to be in the space $\mathrm{Im}(B_\Delta)$, which also implies that $\boldsymbol{\lambda}_k \in \mathrm{Im}(B_\Delta)$ for all $k \geq 1$.

Most of the computational work in each PCG-iteration goes in the application of the dual matrix F and in the application of the preconditioner M^{-1}. While the latter involves the solutions of local Dirichlet problems, the application of the matrix-vector multiplication $F\boldsymbol{\lambda}^k = (F_{\lambda_{RR}} + F_{\lambda_{R\Pi}} \tilde{S}_\Pi^{-1} F_{\lambda_{R\Pi}}^{\mathrm{T}})\boldsymbol{\lambda}^k$ can be implemented in two steps ([1]):

$$\text{S1:} \qquad \boldsymbol{\delta}^k = F_{\lambda_{RR}} \boldsymbol{\lambda}^k = \sum_{s=1}^{N_s} B_R^{(s)} {A_{RR}^{(s)}}^{-1} {B_R^{(s)}}^{\mathrm{T}} \boldsymbol{\lambda}^k,$$

$$\text{S2:} \qquad \boldsymbol{\delta}^k = \boldsymbol{\delta}^k + F_{\lambda_{R\Pi}} \tilde{S}_\Pi^{-1} F_{\lambda_{R\Pi}}^{\mathrm{T}} \boldsymbol{\lambda}^k.$$

Step S1 is easily parallelizable, because it only involves subdomain level computations. Step S2 can be split into the following 3 substeps:

$$\text{S2-a:} \qquad \boldsymbol{\gamma}_a^k = F_{\lambda_{R\Pi}}^{\mathrm{T}} \boldsymbol{\lambda}^k = \sum_{s=1}^{N_s} P_{\Pi}^{(s)\mathrm{T}} \tilde{A}_{R\Pi}^{(s)\ \mathrm{T}} A_{RR}^{(s)\ -1} B_R^{(s)\mathrm{T}} \boldsymbol{\lambda}^k,$$

$$\text{S2-b:} \qquad \text{Solve} \quad \tilde{S}_{\Pi} \boldsymbol{\gamma}_b^k = \boldsymbol{\gamma}_a^k,$$

$$\text{S3-c:} \qquad \boldsymbol{\gamma}_c^k = F_{\lambda_{R\Pi}} \boldsymbol{\gamma}_b^k = \sum_{s=1}^{N_s} B_R^{(s)} A_{RR}^{(s)\ -1} \tilde{A}_{R\Pi}^{(s)} P_{\Pi}^{(s)} \boldsymbol{\gamma}_b^k,$$

where the operators $P_{\Pi}^{(s)}$ are the restriction operators described in Remark 2.2. Steps S2-a and S2-c only involve local computations that can be parallelized at the subdomain level. The product $A_{RR}^{(s)\ -1} B_R^{(s)\mathrm{T}} \boldsymbol{\lambda}^k$ has already been computed in Step S1 and the product $A_{RR}^{(s)\ -1} \tilde{A}_{R\Pi}^{(s)}$ is evaluated only once. Step S2-b is the so called *FETI-DP coarse problem* and can be interpreted as the solution of an auxiliary problem on a coarser mesh (on second level). The coarse matrix $\tilde{S}_{\Pi}$ can be assembled in parallel, using the precomputed products $A_{RR}^{(s)\ -1} \tilde{A}_{R\Pi}^{(s)}$.

2.4 Convergence Analysis

In this section we analyse the convergence of the preconditioned conjugate gradient method applied to the dual problem (2.7) by using the Dirichlet preconditioner (2.11). The analysis uses former results presented in [5, 13, 15], and is here complemented with some additional remarks and explanations.

Let B_{Δ} and $B_{\Delta,D}$ be the continuity operator and its scaled version, respectively. If $\tilde{W}_{\Delta}^h(\Omega)$ is a subspace of $\tilde{W}^h(\Omega)$ spanned by all the basis functions in $\tilde{W}^h(\Omega)$ associated to the dual nodes, a projector

$$P_{\Delta} : \tilde{W}_{\Delta}^h(\Omega) \to \mathrm{Im}(B_{\Delta,D}^{\mathrm{T}}) \subset \tilde{W}_{\Delta}^h(\Omega)$$

can be defined by

$$P_{\Delta} := B_{\Delta,D}^{\mathrm{T}} B_{\Delta}.$$

Proposition 2.8. *The projector P_{Δ} preserves the jumps of any function $\boldsymbol{u}_{\Delta} \in \tilde{W}_{\Delta}^h(\Omega)$, i.e.*

$$B_{\Delta} P_{\Delta} \boldsymbol{u}_{\Delta} = B_{\Delta} \boldsymbol{u}_{\Delta}.$$

Proof. Let two subdomains $\overline{\Omega}^{(s)}$ and $\overline{\Omega}^{(t)}$ share a dual node x. Let the k-th row of B_{Δ} enforce continuity between the two subdomain solutions $\boldsymbol{u}^{(s)}$ and $\boldsymbol{u}^{(t)}$ across x (therefore,

k is such that $k = k(s,t,x) = k(t,s,x)$). Without loss of generality we assume that $s < t$. The continuity operator B_Δ is built such that

$$(B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} = \operatorname{sgn}(t-s)\left(u_\Delta^{(s)}(x) - u_\Delta^{(t)}(x)\right) = \left(u_\Delta^{(s)}(x) - u_\Delta^{(t)}(x)\right),$$

where $u_\Delta^{(s)}$ and $u_\Delta^{(t)}$ are the piecewise polynomial functions built from the weights of $\boldsymbol{u}_\Delta$ corresponding to the dual part of $\overline{\Omega}^{(s)}$ and $\overline{\Omega}^{(t)}$, respectively. Let for a dual node x the functions $\delta_s^\dagger(x)$ be defined as in Eq. (2.9). $\mathcal{N}_x$ contains the indices of the subdomains sharing x and we define for $s \in \mathcal{N}_x$ the set $\mathcal{M}_x^{(s)} := \mathcal{N}_x \backslash \{s\}$. For an arbitrary vector of Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^{n_{cc}}$, we denote by $\left(B_{\Delta,D}^{\mathrm{T}}\boldsymbol{\lambda}\right)_\Delta^{(s)}$ the piecewise polynomial function constructed from the weights of $B_{\Delta,D}^{\mathrm{T}}\boldsymbol{\lambda} \in \tilde{W}_\Delta^h(\Omega)$ corresponding to the dual part of $\overline{\Omega}^{(s)}$, and by the definition of $B_{\Delta,D}$, for any dual node $x \in \partial\Omega^{(s)} \cap \Gamma$ we have

$$\left(B_{\Delta,D}^{\mathrm{T}}\boldsymbol{\lambda}\right)_\Delta^{(s)}(x) = \sum_{r \in \mathcal{M}_x^{(s)}} \operatorname{sgn}(r-s)\delta_r^\dagger(x)\boldsymbol{\lambda}_{k(r,s,x)}.$$

For an arbitrary $\boldsymbol{u}_\Delta \in \tilde{W}_\Delta^h(\Omega)$ we have $P_\Delta \boldsymbol{u}_\Delta \in \tilde{W}_\Delta^h(\Omega)$ and thus

$$\begin{aligned}
(B_\Delta P_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} &= \left[(P_\Delta \boldsymbol{u}_\Delta)_\Delta^{(s)}(x) - (P_\Delta \boldsymbol{u}_\Delta)_\Delta^{(t)}(x)\right] \\
&= \left[\sum_{r \in \mathcal{M}_x^{(s)}} \operatorname{sgn}(r-s)\delta_r^\dagger(x)\,(B_\Delta \boldsymbol{u}_\Delta)_{k(r,s,x)} - \sum_{l \in \mathcal{M}_x^{(t)}} \operatorname{sgn}(l-t)\delta_l^\dagger(x)\,(B_\Delta \boldsymbol{u}_\Delta)_{k(l,t,x)}\right] \\
&= \Bigg[\left(\operatorname{sgn}(t-s)\delta_t^\dagger(x) - \operatorname{sgn}(s-t)\delta_s^\dagger(x)\right)(B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} \\
&\quad + \sum_{r \in \mathcal{M}_x^{(s)} \cap \mathcal{M}_x^{(t)}} \delta_r^\dagger(x) \underbrace{\left(\operatorname{sgn}(r-s)\,(B_\Delta \boldsymbol{u}_\Delta)_{k(r,s,x)} - \operatorname{sgn}(r-t)\,(B_\Delta \boldsymbol{u}_\Delta)_{k(r,t,x)}\right)}_{=:\psi(r,s,t,x)}\Bigg]
\end{aligned}$$

where for the coefficients $\psi(r,s,t,x)$ there are three possible cases:

(1) $\underline{r < s < t:}$

$$\begin{aligned}
\psi(r,s,t,x) &= -(B_\Delta \boldsymbol{u}_\Delta)_{k(r,s,x)} + (B_\Delta \boldsymbol{u}_\Delta)_{k(r,t,x)} \\
&= -\left(u_\Delta^{(r)}(x) - u_\Delta^{(s)}(x)\right) + \left(u_\Delta^{(r)}(x) - u_\Delta^{(t)}(x)\right) \\
&= (B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)}
\end{aligned}$$

(2) $\underline{s < r < t :}$

$$\begin{aligned}\psi(r,s,t,x) &= (B_\Delta \boldsymbol{u}_\Delta)_{k(r,s,x)} + (B_\Delta \boldsymbol{u}_\Delta)_{k(r,t,x)} \\ &= -\left(u_\Delta^{(r)}(x) - u_\Delta^{(s)}(x)\right) + \left(u_\Delta^{(r)}(x) - u_\Delta^{(t)}(x)\right) \\ &= (B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)}\end{aligned}$$

(3) $\underline{s < t < r :}$

$$\begin{aligned}\psi(r,s,t,x) &= (B_\Delta \boldsymbol{u}_\Delta)_{k(r,s,x)} - (B_\Delta \boldsymbol{u}_\Delta)_{k(r,t,x)} \\ &= -\left(u_\Delta^{(r)}(x) - u_\Delta^{(s)}(x)\right) + \left(u_\Delta^{(r)}(x) - u_\Delta^{(t)}(x)\right) \\ &= (B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)}\end{aligned}$$

and therefore we conclude the proof by:

$$\begin{aligned}&(B_\Delta P_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} \\ &\quad= \left[\left(\delta_t^\dagger(x) + \delta_s^\dagger(x)\right)(B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} + \sum_{r \in \mathcal{M}_x^{(s)} \cap \mathcal{M}_x^{(t)}} \delta_r^\dagger(x)\,(B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)}\right] \\ &\quad= (B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)} \sum_{s \in \mathcal{N}_x} \delta_s^\dagger(x) \\ &\quad= (B_\Delta \boldsymbol{u}_\Delta)_{k(s,t,x)}\end{aligned}$$

□

A similar computation shows that the transpose P_Δ^{T} preserves the scaled jumps, i.e.

$$B_{\Delta,D} P_\Delta^{\mathrm{T}} \boldsymbol{u}_\Delta = B_{\Delta,D} \boldsymbol{u}_\Delta.$$

Because the space of Lagrange multipliers is $V = \mathrm{Im}(B_\Delta) = \mathrm{Im}(B_{\Delta,D})$ we have that for any $\boldsymbol{\lambda} \in V$ there is a $\boldsymbol{u}_\Delta \in \tilde{W}_\Delta$ such that $\boldsymbol{\lambda} = B_{\Delta,D} \boldsymbol{u}_\Delta$, and therefore:

$$\boldsymbol{\lambda} = B_{\Delta,D} \boldsymbol{u}_\Delta = B_{\Delta,D} P_\Delta^{\mathrm{T}} \boldsymbol{u}_\Delta = B_{\Delta,D} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}. \tag{2.12}$$

In [1, 15] it is shown that for the three algorithms A, B and C, the projector P_Δ satisfies a stability condition, in particular there is a constant C such that for all $\boldsymbol{u}_\Delta \in \tilde{W}_\Delta$, we have

$$|P_\Delta \boldsymbol{u}_\Delta|_{S_\Delta}^2 \leq C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 |\boldsymbol{u}_\Delta|_{S_\Delta}^2 \tag{2.13}$$

where $|\boldsymbol{u}_\Delta|_{S_\Delta}$ is the norm given by

$$|\boldsymbol{u}_\Delta|_{S_\Delta} = \sqrt{\langle S_\Delta \boldsymbol{u}_\Delta, \boldsymbol{u}_\Delta \rangle},$$

H and h are the characteristic subdomain and mesh sizes, respectively, and C is independent of H and h. We are now ready to state and proof a condition number estimate for the FETI-DP method applied to the Poisson problem (2.1) originally proposed in [1].

Theorem 2.9. *The condition number of the preconditioned dual matrix satisfies*

$$\kappa(M^{-1/2} F M^{-1/2}) \leq C \left(1 + \log\left(\frac{H}{h}\right)\right)^2, \tag{2.14}$$

where C is independent of H and h.

Proof. As $M^{-1/2} F M^{-1/2} = M^{1/2}(M^{-1}F)M^{-1/2}$ the two matrices $M^{-1/2} F M^{-1/2}$ and $M^{-1}F$ have the same eigenvalues and thus, we have to estimate the smallest eigenvalue $\lambda_{min}(M^{-1}F)$ from below and the largest eigenvalue $\lambda_{max}(M^{-1}F)$ from above. For this it is sufficient to show that for all $\boldsymbol{\lambda} \in V$:

$$\langle \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F \leq \langle M^{-1} F \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F \leq C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 \langle \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F$$

Lower bound. By Eq. (2.12) and the Cauchy-Schwarz inequality we have that for all $\boldsymbol{\lambda} \in V$:

$$\begin{aligned}
\langle \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F^2 &= \langle \boldsymbol{\lambda}, B_{\Delta,D} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \rangle_F^2 \\
&= \langle \boldsymbol{\lambda}, B_{\Delta,D} S_\Delta^{1/2} S_\Delta^{-1/2} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \rangle_F^2 \\
&= \langle F\boldsymbol{\lambda}, B_{\Delta,D} S_\Delta^{1/2} S_\Delta^{-1/2} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \rangle^2 \\
&= \langle S_\Delta^{1/2} B_{\Delta,D}^{\mathrm{T}} F\boldsymbol{\lambda}, S_\Delta^{-1/2} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \rangle^2 \\
&\leq \langle S_\Delta^{1/2} B_{\Delta,D}^{\mathrm{T}} F\boldsymbol{\lambda}, S_\Delta^{1/2} B_{\Delta,D}^{\mathrm{T}} F\boldsymbol{\lambda} \rangle \langle S_\Delta^{-1/2} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, S_\Delta^{-1/2} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \rangle \\
&= \langle B_{\Delta,D} S_\Delta B_{\Delta,D}^{\mathrm{T}} F\boldsymbol{\lambda}, F\boldsymbol{\lambda} \rangle \langle B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \\
&= \langle M^{-1} F\boldsymbol{\lambda}, F\boldsymbol{\lambda} \rangle \langle F\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \\
&= \langle M^{-1} F\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F \langle \boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle_F.
\end{aligned}$$

Upper bound. For all $\boldsymbol{\lambda} \in V$ we have $S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda} \in \tilde{W}_\Delta$ and we can use the continuity equation (2.13) for obtaining:

$$\begin{aligned}
\langle M^{-1} F\boldsymbol{\lambda}, \boldsymbol{\lambda}\rangle_F &= \langle M^{-1} F\boldsymbol{\lambda}, F\boldsymbol{\lambda}\rangle \\
&= \langle B_{\Delta,D} S_\Delta B_{\Delta,D}^{\mathrm{T}} B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}\rangle \\
&= \langle B_{\Delta,D}^{\mathrm{T}} B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, B_{\Delta,D}^{\mathrm{T}} B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}\rangle_{S_\Delta} \\
&= |P_\Delta (S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda})|_{S_\Delta}^2 \\
&\leq C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 |S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}|_{S_\Delta}^2 \\
&= C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 \langle S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}\rangle_{S_\Delta} \\
&= C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 \langle B_\Delta S_\Delta^{-1} B_\Delta^{\mathrm{T}} \boldsymbol{\lambda}, \boldsymbol{\lambda}\rangle \\
&= C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 \langle F\boldsymbol{\lambda}, \boldsymbol{\lambda}\rangle \\
&= C \left(1 + \log\left(\frac{H}{h}\right)\right)^2 \langle \boldsymbol{\lambda}, \boldsymbol{\lambda}\rangle_F.
\end{aligned}$$

□

2.5 Numerical Results

In this section we present some numerical results of the FETI-DP method applied to the 2- and 3-dimensional Poisson problem (2.1). When dealing with a 2-dimensional problem it is adequate to define the set of primal nodes to be the set of subdomain vertices. Conversely, if the Poisson problem is solved in a 3-dimensional setting, it turns out that it is advantageous to slightly extend the set of primal nodes and add some additional primal constraints for the convergence rate of the PCG method. As we increase the number of primal degrees of freedom, we increase the size of the coarse problem as well. The idea is to select a small number of additional primal constraints (e.g. equal edge/face averages across the interface) to get a sensible improvement of the convergence rate. All the following numerical results have been carried out by a self developed Matlab code.

2-dimensional Poisson problem

Let the computational domain $\Omega = (0,1)^2$ be decomposed into $N_s = N \times N$ square subdomains with side length $H = 1/N$. We use a structured triangular mesh of size h. On the left and bottom part, homogeneous Dirichlet boundary conditions are imposed,

while on the remaining parts homogeneous Neumann conditions are used (see Figure 2.4). The diffusion coefficient $\rho = 1$ is kept constant within the domain Ω and for this

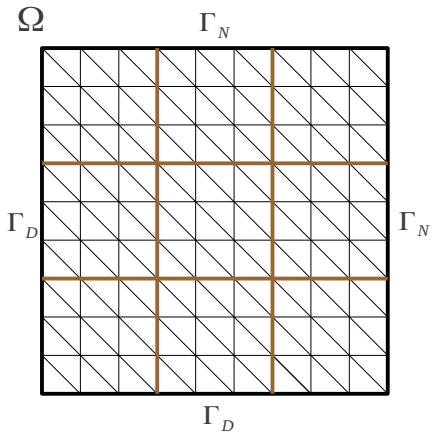


FIGURE 2.4: Computational domain Ω being the unit square decomposed into $N_s = N \times N$ square subdomains.

first implementation we use the identity matrices as scaling weights. In order to examine the scalability of the method, we keep the ratio H/h fixed and increase the number of subdomains. The number of PCG iterations for different numbers of subdomains are illustrated in Table 2.1. We observe that the upper bound of iterations is reached

H	N_s	H/h	Dofs/Subd	Dofs	N_{Iter}
1/5	25	10	121	2'601	8
1/6	36	10	121	3'721	9
1/7	49	10	121	5'041	12
1/8	64	10	121	6'561	13
1/9	81	10	121	8'281	15
1/10	100	10	121	10'201	16
1/11	121	10	121	12'321	16
1/12	144	10	121	14'641	16
1/13	169	10	121	17'161	16
1/14	196	10	121	19'881	16
1/15	255	10	121	22'801	16

TABLE 2.1: Number of PCG iterations using FETI-DP for a 2-dimensional problem.

very quickly and therefore, the algorithm has a very good parallel scalability for this 2-dimensional model problem.

3-dimensional Poisson problem

For a 3-dimensional subdomain $\Omega^{(s)} \subset \mathbb{R}^3$ we denote by $\mathcal{V}_i^{(s)}$, $\mathcal{E}_j^{(s)}$ and $\mathcal{F}_k^{(s)}$ the subdomain vertices, edges and faces in $\partial\Omega^{(s)} \backslash \Gamma_D$, respectively. For an illustration of the edges and the faces in a cube with $\Gamma_D = \partial\Omega$, being subdivided into 8 subdomains see Figure 2.5. Three possible sets of primal nodes in the 3-dimensional case are suggested

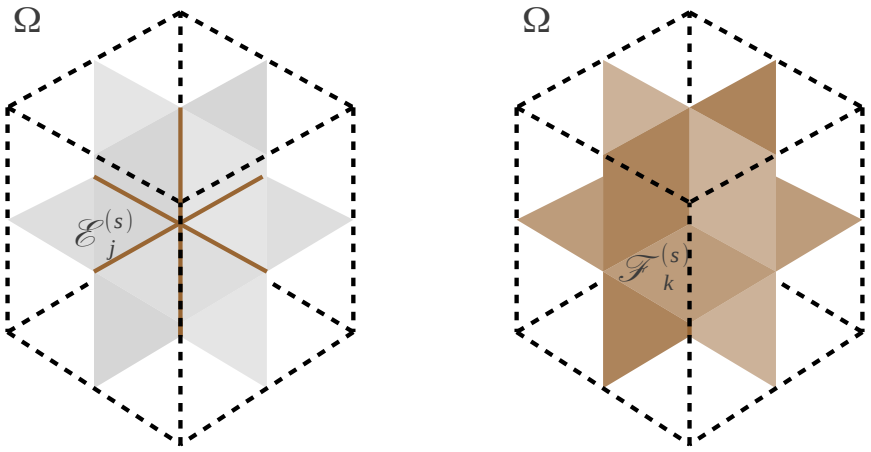


FIGURE 2.5: Illustration of the edges (left) and faces (right) in 3-dimensional cube subdivided into 8 subdomains.

in [13, 15] and are given by:

Algorithm 2.10 (Algorithm A). *The subspace $\tilde{W}$ of W is the space of functions being continuous across all the subdomain vertices $\mathcal{V}_i^{(s)}$, $\forall s = 1, \ldots, N_s$.*

Algorithm 2.11 (Algorithm B). *The subspace $\tilde{W}$ of W is the space of functions being continuous across all the subdomain vertices $\mathcal{V}_i^{(s)}$, $\forall s = 1, \ldots, N_s$, and, in addition, they have the same average value in the interior of each subdomain edge $\mathcal{E}_j^{(s)}$.*

Algorithm 2.12 (Algorithm C). *The subspace $\tilde{W}$ of W is the space of functions being continuous across all the subdomain vertices $\mathcal{V}_i^{(s)}$, $\forall s = 1, \ldots, N_s$, and, in addition, they have the same average value in the interior of each subdomain face $\mathcal{F}_k^{(s)}$.*

Furthermore, we will consider two different ways for the implementation of algorithms B and C. The first (cf. [13]), is to perform a change of basis such that the new basis restricted to each primal edge or primal face consists in average-free basis functions plus one basis function being constant in the interior of the edge or the face (see Example 2.3).

Example 2.3 (Change of basis). *Consider an edge $\mathcal{E}_i^{(s)}$ and let $\{\varphi_1^{(s)}, \ldots, \varphi_n^{(s)}\}$ be the set of local finite element basis functions associated to the nodes lying on the interior of*

$\mathcal{E}_i^{(s)}$. *We define a boolean operator* $T_{\mathcal{E}_i^{(s)}} \in \{0,1,-1\}^{n\times n}$ *such that:*

$$T_{\mathcal{E}_i^{(s)}} = \left(\begin{array}{ccc:c} & & & -1 \\ & I_{n-1} & & \vdots \\ & & & -1 \\ \hdashline 1 & \cdots & 1 & 1 \end{array}\right),$$

where I_{n-1} *is the* $n-1\times n-1$ *identity matrix. Then, the set of new basis functions* $\{\tilde{\varphi}_1^{(s)},\dots,\tilde{\varphi}_n^{(s)}\}$ *is given by:*

$$\tilde{\varphi}_j^{(s)} = \sum_{k=1}^{n} \left(T_{\mathcal{E}_i^{(s)}}\right)_{jk} \varphi_k^{(s)}, \qquad \forall j = 1,\dots,n.$$

If an edge $\mathcal{E}$ *is shared by the two subdomains* $\Omega^{(s)}$ *and* $\Omega^{(t)}$*, in addition to the weights associated to the vertex functions, we require the weights in the two subdomains associated to the average basis function to be equal. If* $T^{(s)}$ *is an operator performing all the changes of basis in the subdomain* $\Omega^{(s)}$*, the local matrix* $A^{(s)}$ *and the local vector* $\boldsymbol{f}^{(s)}$ *become*

$$\overline{A}^{(s)} = T^{(s)} A^{(s)} {T^{(s)}}^{\mathrm{T}}, \qquad \overline{\boldsymbol{f}}^{(s)} = T^{(s)} \boldsymbol{f}^{(s)}.$$

For an illustration of the change of basis for an edge, see Figure 2.6. The same procedure

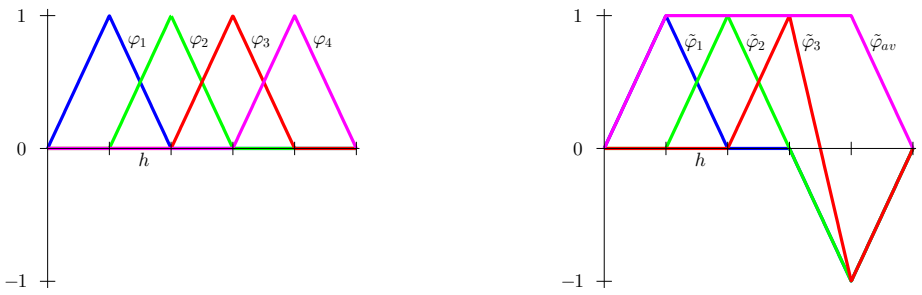


FIGURE 2.6: Illustration of the change of basis. (Left) The standard piecewise linear basis functions restricted to an edge. (Right) The new basis consisting of 3 average free basis functions and one primal basis function representing the average.

reads in the case of face averages as additional primal constraints.

The second way (cf. [15]), relies in imposing direct average equalities by an additional set of nonredundant Lagrange multipliers. Such average conditions can be expressed through a boolean matrix Q_R, such that the average equality reads:

$$Q_R \boldsymbol{u}_R = 0.$$

By using Lagrange multipliers to enforce average equalities, the saddle point system becomes:

$$\begin{pmatrix} A_{RR} & \tilde{K}_{R\Pi} & B_R^{\mathrm{T}} \\ \tilde{K}_{R\Pi}^{\mathrm{T}} & \tilde{K}_{\Pi\Pi} & 0 \\ B_R & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_R \\ \tilde{\boldsymbol{v}}_\Pi \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_R \\ \tilde{\boldsymbol{g}}_\Pi \\ 0 \end{pmatrix}, \tag{2.15}$$

where

$$\tilde{K}_{R\Pi} = \begin{pmatrix} \tilde{A}_{R\Pi} & Q_R^{\mathrm{T}} \end{pmatrix}, \quad \tilde{K}_{\Pi\Pi} = \begin{pmatrix} \tilde{A}_{\Pi\Pi} & 0 \\ 0 & 0 \end{pmatrix}, \quad \tilde{\boldsymbol{v}}_\Pi = \begin{pmatrix} \tilde{\boldsymbol{u}}_\Pi \\ \boldsymbol{\mu} \end{pmatrix}, \quad \tilde{\boldsymbol{g}}_\Pi = \begin{pmatrix} \tilde{\boldsymbol{f}}_\Pi, \\ 0 \end{pmatrix}$$

and $\boldsymbol{\mu}$ is the additional set of Lagrange multipliers enforcing the average constraints.

We have applied the FETI-DP technique using algorithms A, B and C to the Poisson problem (2.1). We consider a unit cube $\Omega = (0,1)^3$ decomposed into $N_s := N \times N \times N$ cubic subdomains with side length $H = 1/N$ and a structured tetrahedral mesh of size h (see Figure 2.7). The diffusion coefficients ρ_s are kept constant within each subdomain

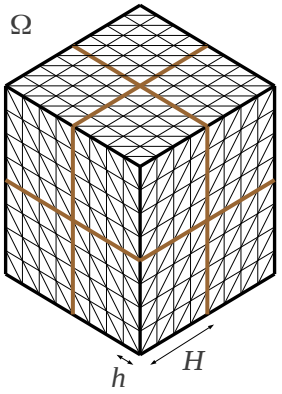


FIGURE 2.7: Computational domain Ω being the unit cube decomposed into $N_s := N \times N \times N$ cubic subdomains with side length $H = 1/N$.

and vary between 1 and 10^4 in a 3-dimensional checkerboard pattern. On the front, left, and bottom part, homogeneous Dirichlet boundary conditions are imposed, while on the remaining parts homogeneous Neumann conditions are used. In order to examine the scalability of the method we keep the ratio H/h fixed, increase the number of subdomains and report the number of PCG iterations for different numbers of subdomains. The stopping condition for the PCG method is based on the relative reduction of the initial residual by a factor of 10^{-7} (Table 2.2) or 10^{-10} (Table 2.2). The number of iterations are the same regardless if the additional constraints are imposed by a change of basis or by Lagrange multipliers. We can see in both tables that the algorithms B and C have a

H	N_s	H/h	Dofs/Subd	Dofs	N_{Iter}		
					Alg. A	Alg. B	Alg. C
1/2	8	4	125	729	11	12	13
1/3	27	4	125	2'197	23	14	14
1/4	64	4	125	4'913	30	16	14
1/5	125	4	125	9'261	41	17	15
1/6	216	4	125	15'625	44	17	15
1/7	343	4	125	24'389	48	17	15
1/8	512	4	125	35'937	48	17	15

TABLE 2.2: Number of PCG iterations with tolerance tol $= 10^{-7}$ using FETI-DP for a 3-dimensional problem with Algorithms A, B and C.

good parallel scalability for our model problem and the number of degrees of freedoms per subdomain considered.

H	N_s	H/h	Dofs/Subd	Dofs	N_{Iter}		
					Alg. A	Alg. B	Alg. C
1/2	8	4	125	729	15	16	18
1/3	27	4	125	2'197	31	19	20
1/4	64	4	125	4'913	46	21	20
1/5	125	4	125	9'261	57	23	21
1/6	216	4	125	15'625	61	24	20
1/7	343	4	125	24'389	68	24	20
1/8	512	4	125	35'937	69	24	20

TABLE 2.3: Number of PCG iterations with tolerance tol $= 10^{-10}$ using FETI-DP for a 3-dimensional problem with Algorithms A, B and C.

Chapter 3

FETI-DP applied to Navier-Stokes Equations

3.1 Strong and Weak Form of the Navier-Stokes Equations

In Chapter 2 we have presented the FETI-DP domain decomposition method applied to the Poisson equation. Now, we extend the method to the time dependent Navier-Stokes equations ([10, 11]). The Navier-Stokes equations for an incompressible fluid describe the motion of a fluid with constant density ρ in a domain $\Omega \subset \mathbb{R}^d$, with $d = 2, 3$ and $|\Omega| < +\infty$. For $T > 0$, the Navier-Stokes equations read:

$$\begin{cases} \dfrac{\partial \boldsymbol{u}}{\partial t} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nabla p = \boldsymbol{f} & \text{in } (0,T) \times \Omega \\ \nabla \cdot \boldsymbol{u} = 0 & \text{in } (0,T) \times \Omega \end{cases} \tag{3.1}$$

being $\boldsymbol{u} = \boldsymbol{u}(t, \boldsymbol{x})$ the velocity field, $p = p(t, \boldsymbol{x})$ the pressure divided by the density ρ, $\nu = \dfrac{\mu}{\rho}$ the kinematic viscosity and $\boldsymbol{f} = \boldsymbol{f}(t, \boldsymbol{x})$ the ratio between the forcing term per unit volume and the density. Furthermore, we consider the following initial condition

$$\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}_0(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Omega, \tag{3.2}$$

where $\boldsymbol{u}_0$ is a divergence free velocity field. The boundary of Ω is split into the subsets Γ_D and Γ_N, where Dirichlet and Neumann conditions apply, respectively. The subdivision of the boundary is such that $\overset{\circ}{\Gamma}_D \cap \overset{\circ}{\Gamma}_N = \emptyset$ and $\partial\Omega = \overline{\Gamma_D \cup \Gamma_N}$. With respect to Eq.

(3.1) the following boundary conditions are considered:

$$\begin{cases} \boldsymbol{u}(t,\boldsymbol{x}) = \boldsymbol{\varphi}(t,\boldsymbol{x}) & \text{on } \Gamma_D \\ \left(\nu\dfrac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} - p\boldsymbol{n}\right)(t,\boldsymbol{x}) = \boldsymbol{\psi}(t,\boldsymbol{x}) & \text{on } \Gamma_N \end{cases} \tag{3.3}$$

where $\boldsymbol{\varphi}$ and $\boldsymbol{\psi}$ are given and sufficiently smooth functions and $\boldsymbol{n}$ is the outward unit vector normal to $\partial\Omega$. We write the boundary value problem (3.1) - (3.3) in variational form. The test function spaces are chosen as

$$V = \left[H^1_{\Gamma_D}(\Omega)\right]^d, \quad \text{and} \quad Q = \begin{cases} L^2(\Omega) & \text{if } \Gamma_N \neq \emptyset \\ L^2_0(\Omega) & \text{if } \Gamma_N = \emptyset \end{cases}$$

where $L^2_0(\Omega) = \{q \in L^2(\Omega) \ : \ \displaystyle\int_\Omega q d\Omega = 0\}$. The reason for this differentiation is that in the case of a pure Dirichlet problem (i.e., if $\Gamma_D = \partial\Omega$), the pressure p appears only through its gradient (see Eq. (3.1) and (3.3)) and therefore, for any $c \in \mathbb{R}$, $\nabla p = \nabla(p+c)$ which means that the pressure is unique only up to a constant. In addition, if $\Gamma_D = \partial\Omega$, the Dirichlet data $\boldsymbol{\varphi}$ has to fulfil the compatibility condition

$$0 = \int_\Omega \nabla \cdot \boldsymbol{u} d\Omega = \int_{\partial\Omega} \boldsymbol{\varphi} \cdot \boldsymbol{n} d\gamma.$$

The weak formulation of the Navier-Stokes equation for an incompressible fluid reads: find $\boldsymbol{u} \in L^2\big((0,T); \left[H^1(\Omega)\right]^d\big)$ and $p \in L^2\big((0,T); Q\big)$ such that

$$\begin{cases} \displaystyle\int_\Omega \frac{\partial \boldsymbol{u}}{\partial t} \cdot \boldsymbol{v} d\Omega + \nu \int_\Omega \nabla \boldsymbol{u} : \nabla \boldsymbol{v} d\Omega + \int_\Omega [(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}] \cdot \boldsymbol{v} d\Omega - \int_\Omega p \nabla \cdot \boldsymbol{v} d\Omega \\ \qquad = \displaystyle\int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} d\Omega + \int_{\Gamma_N} \boldsymbol{\psi} \cdot \boldsymbol{v} d\Gamma \qquad \forall \boldsymbol{v} \in V, \\ \displaystyle\int_\Omega q \nabla \cdot \boldsymbol{u} d\Omega = 0 \qquad \forall q \in Q, \end{cases} \tag{3.4}$$

with $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}_0(\boldsymbol{x})$ and $\boldsymbol{u}|_{\Gamma_D} = \boldsymbol{\varphi}$. In this context, the tensor product between two tensors $A, B \in \mathbb{R}^{d\times d}$ is $A : B = \displaystyle\sum_{i,j=1}^{d} A_{ij}B_{ij}$.

3.2 Time Discretization Scheme

The time interval $(0,T)$ is subdivided into N intervals of length $\Delta t = T/N$, so that $t_n = n\Delta t$, for $n = 0, \ldots, N$. We approximate in time the weak formulation (3.4) by a *Backward Differentiation Formula* of order p (*BDFp*) (see e.g. [3]). Depending on the order p of the BDF scheme, for $n \geq n_0$ the time derivative of $\boldsymbol{u}$ at t_{n+1} can be

approximated by

$$\frac{\partial \boldsymbol{u}}{\partial t}(t_{n+1}, \boldsymbol{x}) \approx \frac{\alpha}{\Delta t}\boldsymbol{u}(t_{n+1}, \boldsymbol{x}) - \frac{1}{\Delta t}\boldsymbol{u}_{BDF}(\boldsymbol{x}).$$

Henceforth, for $n = 0, \ldots, N$ we denote the velocity field $\boldsymbol{u}$ at time t_n by $\boldsymbol{u}^n(\boldsymbol{x}) = \boldsymbol{u}(t_n, \boldsymbol{x})$. The constants α and n_0 as well as $\boldsymbol{u}_{BDF}$ for the different schemes are given by

$$\begin{aligned}
&\text{BDF1:} && \boldsymbol{u}_{BDF}(\boldsymbol{x}) = \boldsymbol{u}^n(\boldsymbol{x}),\ \alpha = 1,\ n_0 = 0\\
&\text{BDF2:} && \boldsymbol{u}_{BDF}(\boldsymbol{x}) = 2\boldsymbol{u}^n(\boldsymbol{x}) - \frac{1}{2}\boldsymbol{u}^{n-1}(\boldsymbol{x}),\ \alpha = \frac{3}{2},\ n_0 = 1\\
&\text{BDF3:} && \boldsymbol{u}_{BDF}(\boldsymbol{x}) = 3\boldsymbol{u}^n(\boldsymbol{x}) - \frac{3}{2}\boldsymbol{u}^{n-1}(\boldsymbol{x}) + \frac{1}{3}\boldsymbol{u}^{n-2}(\boldsymbol{x}),\ \alpha = \frac{3}{2},\ n_0 = 2\\
&\text{BDF4:} && \boldsymbol{u}_{BDF}(\boldsymbol{x}) = 4\boldsymbol{u}^n(\boldsymbol{x}) - 3\boldsymbol{u}^{n-1}(\boldsymbol{x}) + \frac{4}{3}\boldsymbol{u}^{n-2}(\boldsymbol{x}) - \frac{1}{4}\boldsymbol{u}^{n-3}(\boldsymbol{x}),\ \alpha = \frac{25}{12},\ n_0 = 3
\end{aligned}$$

The Navier-Stokes equations are nonlinear due to the convective term $(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}$. In this work we consider a fully explicit approximation of this term as follows:

$$(\boldsymbol{u}^{n+1}\cdot\nabla)\boldsymbol{u}^{n+1} \approx (\boldsymbol{u}^\star\cdot\nabla)\boldsymbol{u}^\star$$

where, with respect to the order p of the BDF scheme used, $\boldsymbol{u}^\star$ reads:

$$\begin{aligned}
&\text{BDF1:} && \boldsymbol{u}^\star(\boldsymbol{x}) = \boldsymbol{u}^n(\boldsymbol{x})\\
&\text{BDF2:} && \boldsymbol{u}^\star(\boldsymbol{x}) = 2\boldsymbol{u}^n(\boldsymbol{x}) - \boldsymbol{u}^{n-1}(\boldsymbol{x})\\
&\text{BDF3:} && \boldsymbol{u}^\star(\boldsymbol{x}) = 3\boldsymbol{u}^n(\boldsymbol{x}) - 3\boldsymbol{u}^{n-1}(\boldsymbol{x}) + \boldsymbol{u}^{n-2}(\boldsymbol{x})\\
&\text{BDF4:} && \boldsymbol{u}^\star(\boldsymbol{x}) = 4\boldsymbol{u}^n(\boldsymbol{x}) - 6\boldsymbol{u}^{n-1}(\boldsymbol{x}) + 4\boldsymbol{u}^{n-2}(\boldsymbol{x}) - \boldsymbol{u}^{n-3}(\boldsymbol{x}).
\end{aligned}$$

Supposing to know a sufficient regular lifting function $\boldsymbol{\Phi} : \overline{\Omega} \to \mathbb{R}^d$ such that $\boldsymbol{\Phi}|_{\Gamma_D} = \boldsymbol{\varphi}$, the problem consists in finding $(\boldsymbol{u}^{n+1}, p^{n+1}) \in V \times Q$, solution of

$$\begin{cases} a(\boldsymbol{u}^{n+1}, \boldsymbol{v}) + b(\boldsymbol{v}, p) = F^{n+1}(\boldsymbol{v}) - a(\boldsymbol{\Phi}, \boldsymbol{v}) & \forall \boldsymbol{v} \in V,\\ b(\boldsymbol{u}^{n+1}, q) = -b(\boldsymbol{\Phi}, q) & \forall q \in Q, \end{cases} \tag{3.5}$$

with the bilinear forms $a : V \times V \to \mathbb{R}$ and $b : V \times Q \to \mathbb{R}$ defined as:

$$a(\boldsymbol{u}, \boldsymbol{v}) = \frac{\alpha}{\Delta t}\int_\Omega \boldsymbol{u}\cdot\boldsymbol{v}d\Omega + \nu\int_\Omega \nabla\boldsymbol{u} : \nabla\boldsymbol{v}d\Omega \tag{3.6}$$

$$b(\boldsymbol{u}, p) = -\int_\Omega p\nabla\cdot\boldsymbol{u}d\Omega, \tag{3.7}$$

and the linear functional $F^{n+1} : V \to \mathbb{R}$:

$$F^{n+1}(\boldsymbol{v}) = \int_\Omega \boldsymbol{f}^{n+1}\cdot\boldsymbol{v} + \int_{\Gamma_N}\boldsymbol{\psi}^{n+1}\cdot\boldsymbol{v}d\Gamma + \int_\Omega\left[\frac{1}{\Delta t}\boldsymbol{u}_{BDF} - (\boldsymbol{u}^\star\cdot\nabla)\boldsymbol{u}^\star\right]\cdot\boldsymbol{v}d\Omega. \tag{3.8}$$

3.3 Problem Setting and Matrix Equation

A well known result ([10, 11]) from the analysis of the Navier-Stokes problem is, that if the discrete space of test functions V_h is not rich enough with respect to the discrete pressure test function space Q_h, the bilinear form $b(\cdot,\cdot)$ may no longer satisfy the compatibility condition (see Theorem A.1) and therefore the problem (3.4) is not well posed. It is possible to use a stabilization technique ([10, 11]), which allows for example piecewise linear basis functions for the pressure and velocity unknowns. For a suitable bilinear form $c_h : Q \times Q \to \mathbb{R}$ and a family of linear functionals $G_h^n : Q \to \mathbb{R}$ the weak formulation of the stabilized Navier-Stokes equation for an incompressible fluid is to find $(\boldsymbol{u}^{n+1}, p^{n+1}) \in V \times Q$ such that

$$\begin{cases} a(\boldsymbol{u}^{n+1}, \boldsymbol{v}) + b(\boldsymbol{v}, p) = F^{n+1}(\boldsymbol{v}) - a(\boldsymbol{\Phi}, \boldsymbol{v}) & \forall \boldsymbol{v} \in V, \\ b(\boldsymbol{u}^{n+1}, q) - c_h(p, q) = G_h(q) - b(\boldsymbol{\Phi}, q) & \forall q \in Q, \end{cases} \tag{3.9}$$

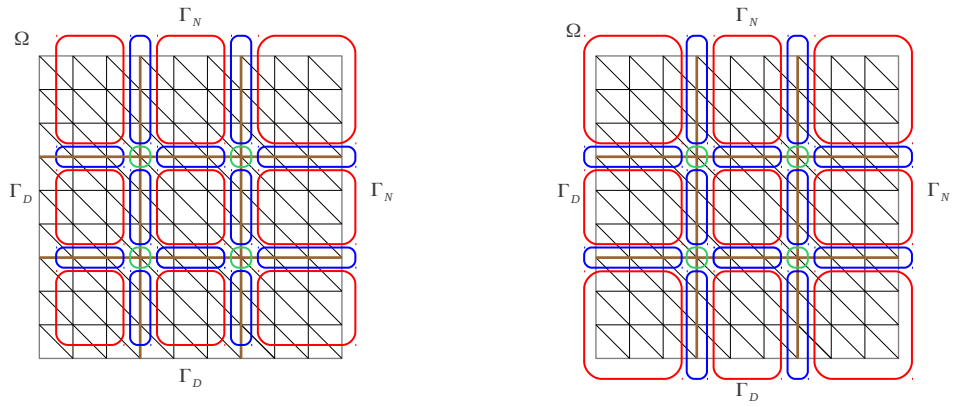
where we use the same notations introduced in Eq. (3.6)-(3.8).

We aim at solving Eq. (3.9) by a FETI-DP method ([10, 15]). The computational domain Ω is subdivided into N_s nonoverlapping subdomains $\{\Omega^{(s)},\ s = 1, \ldots, N_s\}$. The interface Γ and the FE-space $\tilde{W}^h(\Omega)$ are defined as in the Chapter 2 (see Eq. (2.5)), and we set the finite dimensional velocity space $\tilde{W}_v^h(\Omega)$ as

$$\tilde{W}_v^h(\Omega) := \left[\tilde{W}^h(\Omega)\right]^d.$$

Thanks to the lifting function $\boldsymbol{\Phi}$ representing the Dirichlet data on Γ_D, the weights of the degrees of freedom of $\tilde{W}_v^h(\Omega)$ associated to the nodes lying on Γ_D vanishes. The pressure FE-space $\tilde{W}_p^h(\Omega)$ is an extension of $\tilde{W}^h(\Omega)$ by degrees of freedom associated to the nodes on Γ_D which are added either to the set of dual dofs or to the set of internal dofs, depending on whether the associated nodes lie on the interface Γ or not. We use the subscripts I, Δ and Π to denote the internal, the dual and the primal dofs respectively and henceforth we add subscripts u or p for indicating either the velocity or pressure space. We recall that the set of primal nodes is the same in the velocity and in the pressure space (see Figure 3.1 for an example of the node assignment in a 2-dimensional domain.).

The continuity operators $B_{\Delta,u}$ and $B_{\Delta,p}$ are built in the same way as described in chapter 2. The operator $B_{\Delta,u}$, acting on the dual part of the velocity dofs, is the same continuity operator (repeated for each of the d components) introduced in chapter 2, while $B_{\Delta,p}$ is the continuity operator of chapter 2 extended by the continuity conditions associated to the dual nodes lying on Γ_D. In the case of a 3-dimensional domain Ω, we use additional



(A) The dofs of the velocity space are grouped into primal dofs Π_u (green), dual dofs Δ_u (blue) and internal dofs I_u (red)

(B) The dofs in the pressure space are grouped into primal dofs Π_p (green), dual dofs Δ_p (blue) and internal dofs I_p (red)

FIGURE 3.1: Assignment of the velocity and the pressure dofs in a 2-dimensional setting.

Lagrange multipliers in order to enforce a nonredundant set of average constraints on the pressure and on each vectorial component of the velocity, either across the edges or the faces (see Algorithm 2.11 and 2.12). For $n = n_0, \dots, N-1$ and a given $\boldsymbol{u}^n$, the matrix form of the FETI-DP domain decomposition method applied to the Navier-Stokes equation for an incompressible fluid reads:

$$\begin{pmatrix} K_{RR} & K_{R\Pi} & B_R^{\mathrm{T}} \\ K_{R\Pi}^{\mathrm{T}} & K_{\Pi\Pi} & 0 \\ B_R & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_R \\ \boldsymbol{v}_\Pi \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_R \\ \boldsymbol{h}_\Pi \\ 0 \end{pmatrix} \tag{3.10}$$

with

$$K_{RR} = \begin{pmatrix} A_{I_uI_u} & A_{I_u\Delta_u} & B_{I_uI_p} & B_{I_u\Delta_p} \\ A_{I_u\Delta_u}^{\mathrm{T}} & A_{\Delta_u\Delta_u} & B_{\Delta_uI_p} & B_{\Delta_u\Delta_p} \\ B_{I_uI_p}^{\mathrm{T}} & B_{\Delta_uI_p}^{\mathrm{T}} & -C_{I_pI_p} & -C_{I_p\Delta_p} \\ B_{I_u\Delta_p}^{\mathrm{T}} & B_{\Delta_u\Delta_p}^{\mathrm{T}} & -C_{I_p\Delta_p}^{\mathrm{T}} & -C_{\Delta_p\Delta_p} \end{pmatrix}, K_{\Pi\Pi} = \begin{pmatrix} A_{\Pi_u\Pi_u} & B_{\Pi_u\Pi_p} & 0 & 0 \\ B_{\Pi_u\Pi_p}^{\mathrm{T}} & -C_{\Pi_p\Pi_p} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$K_{R\Pi} = \begin{pmatrix} A_{I_u\Pi_u} & B_{I_u\Pi_p} & 0 & 0 \\ A_{\Delta_u\Pi_u} & B_{\Delta_u\Pi_p} & Q_{\Delta,u}^{\mathrm{T}} & 0 \\ B_{\Pi_uI_p}^{\mathrm{T}} & -C_{I_p\Pi_p} & 0 & 0 \\ B_{\Pi_u\Delta_p}^{\mathrm{T}} & -C_{\Delta_p\Pi_p} & 0 & Q_{\Delta,p}^{\mathrm{T}} \end{pmatrix}, \quad B_R = \begin{pmatrix} 0 & B_{\Delta,u} & 0 & 0 \\ 0 & 0 & 0 & B_{\Delta,p} \end{pmatrix}$$

$$\boldsymbol{h}_R = \begin{pmatrix} \boldsymbol{f}_{I_u} \\ \boldsymbol{f}_{\Delta_u} \\ \boldsymbol{g}_{I_p} \\ \boldsymbol{g}_{\Delta_p} \end{pmatrix}, \quad \boldsymbol{h}_\Pi = \begin{pmatrix} \boldsymbol{f}_{\Pi_u} \\ \boldsymbol{g}_{\Pi_p} \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{v}_R = \begin{pmatrix} \boldsymbol{u}_{I_u} \\ \boldsymbol{u}_{\Delta_u} \\ \boldsymbol{p}_{I_p} \\ \boldsymbol{p}_{\Delta_p} \end{pmatrix} \quad \text{and} \quad \boldsymbol{v}_\Pi = \begin{pmatrix} \boldsymbol{u}_{\Pi_u} \\ \boldsymbol{p}_{\Pi_p} \\ \boldsymbol{\mu}_u \\ \boldsymbol{\mu}_p \end{pmatrix},$$

where $\boldsymbol{u} = (\boldsymbol{u}_{I_u}, \boldsymbol{u}_{\Delta_u}, \boldsymbol{u}_{\Pi_u})^{\mathrm{T}} \in \tilde{W}_u^h(\Omega)$ and $\boldsymbol{p} = (\boldsymbol{p}_{I_p}, \boldsymbol{p}_{\Delta_p}, \Pi_p)^{\mathrm{T}} \in \tilde{W}_p^h(\Omega)$. The variables $\boldsymbol{v}_R$ and $\boldsymbol{v}_\Pi$ in (3.10) are eliminated to solve a dual problem of the form

$$F\boldsymbol{\lambda} = \boldsymbol{d}, \tag{3.11}$$

where

$$\begin{aligned}
F &= F_{\lambda_{RR}} + F_{\lambda_{R\Pi}} S_\Pi^{-1} F_{\lambda_{R\Pi}}^{\mathrm{T}}, \\
\boldsymbol{d} &= B_R K_{RR}^{-1} \boldsymbol{h}_R + F_{\lambda_{R\Pi}} S_\Pi^{-1} \left(K_{R\Pi}^{\mathrm{T}} K_{RR}^{-1} \boldsymbol{h}_R - \boldsymbol{h}_\Pi \right), \\
S_\Pi &= K_{\Pi\Pi} - K_{R\Pi}^{\mathrm{T}} K_{RR}^{-1} K_{R\Pi}, \\
F_{\lambda_{RR}} &= B_R K_{RR}^{-1} B_R^{\mathrm{T}}, \\
F_{\lambda_{R\Pi}} &= B_R K_{RR}^{-1} K_{R\Pi}.
\end{aligned}$$

The bilinear form $a(\cdot,\cdot)$ defined in Eq. (3.6) is symmetric and coercive, because for a function $\boldsymbol{u} : \Omega \to \mathbb{R}^d$, $a(\boldsymbol{u},\boldsymbol{u})$ is a scaling of $||\boldsymbol{u}||_{[H^1]^d}$ where $||\cdot||_{[H^1]^d}$ denotes the usual norm defined on the space $[H^1(\Omega)]^d$ ([6]). Therefore, the matrix

$$A = \begin{pmatrix} A_{I_u I_u} & A_{I_u \Delta_u} & A_{I_u \Pi_u} \\ A_{I_u \Delta_u}^{\mathrm{T}} & A_{\Delta_u \Delta_u} & A_{\Delta_u \Pi_u} \\ A_{I_u \Pi_u}^{\mathrm{T}} & A_{\Delta_u \Pi_u}^{\mathrm{T}} & A_{\Pi_u \Pi_u} \end{pmatrix},$$

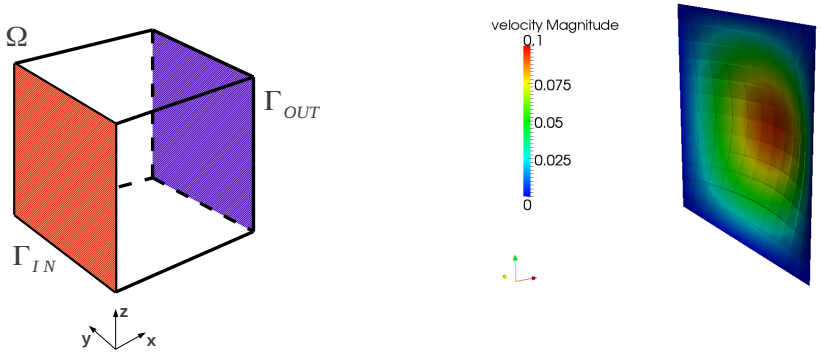
is symmetric and positive definite. One can use similar arguments as in the proof of Proposition 2.7 or as in Chapter 4 of [7] to show that the dual matrix F is symmetric and positive definite in the space $\mathrm{Im}(B_\Delta)$. The dual problem 3.11 is solved by a preconditioned conjugate gradient method, using an initial guess $\boldsymbol{\lambda} \in \mathrm{Im}(B_\Delta)$ and the preconditioner

$$M^{-1} = \sum_{s=1}^{N_s} \begin{pmatrix} B_{\Delta,D,u}^{(s)} \\ B_{\Delta,D,p}^{(s)} \end{pmatrix} S_\Delta^{(s)} \begin{pmatrix} B_{\Delta,D,u}^{(s)\,\mathrm{T}} & B_{\Delta,D,p}^{(s)\,\mathrm{T}} \end{pmatrix}. \tag{3.12}$$

The local Schur complement $S_\Delta^{(s)}$ for $s = 1, \ldots, N_s$ reads

$$\begin{aligned}
S_\Delta^{(s)} = &\begin{pmatrix} A_{\Delta_u \Delta_u}^{(s)} & B_{\Delta_u \Delta_p}^{(s)} \\ B_{\Delta_u \Delta_p}^{(s)\,\mathrm{T}} & -C_{\Delta_p \Delta_p}^{(s)} \end{pmatrix} \\
&- \begin{pmatrix} A_{I_u \Delta_u}^{(s)\,\mathrm{T}} & B_{\Delta_u I_p}^{(s)\,\mathrm{T}} \\ B_{I_u \Delta_p}^{(s)\,\mathrm{T}} & -C_{I_p \Delta_p}^{(s)\,\mathrm{T}} \end{pmatrix} \begin{pmatrix} A_{I_u I_u}^{(s)} & B_{I_u I_p}^{(s)} \\ B_{I_u I_p}^{(s)\,\mathrm{T}} & -C_{I_p I_p}^{(s)} \end{pmatrix}^{-1} \begin{pmatrix} A_{I_u \Delta_u}^{(s)} & B_{I_u \Delta_p}^{(s)} \\ B_{\Delta_u I_p}^{(s)\,\mathrm{T}} & -C_{I_p \Delta_p}^{(s)} \end{pmatrix}.
\end{aligned}$$

The maps $B_{\Delta,D,u}^{(s)}$ and $B_{\Delta,D,p}^{(s)}$ are the restrictions of $B_{\Delta,D,u}$ and $B_{\Delta,D,p}$ to the subdomain $\Omega^{(s)}$, scaled by the weights $\delta_s^\dagger(x)$ indicating the number of subdomains sharing a dual node (see Eq. (2.9) with $\rho \equiv 1$).



(A) Inflow and outflow faces in the unit cube.

(B) Sinusoidal shaped Dirichlet data on the inflow face at $t = 0.1s$.

FIGURE 3.2: Visualization of the boundaries and the inflow boundary conditions.

3.4 Numerical Results

We present some numerical results of the FETI-DP method applied to the incompressible Navier-Stokes equation in the unit cube $\Omega = (0,1)^3$. The domain is decomposed in $N_s := N \times N \times N$ cubic subdomains of side length $H = 1/N$, and we use a structured tetrahedral mesh (see Figure 2.7). On the inflow face:

$$\Gamma_{IN} := \{\boldsymbol{x} \in \overline{\Omega} \ : \ x_1 = 0\}$$

a Dirichlet boundary condition given by

$$\boldsymbol{\Phi}(t, \boldsymbol{x}) = \begin{pmatrix} \tau(t)\sin(\pi x_2)\sin(\pi x_3) \\ 0 \\ 0 \end{pmatrix}, \qquad \text{with} \quad \tau(t) = \begin{cases} t & \text{if } \ 0 \leq t \leq 0.1 \\ 0.1 & \text{otherwise} \end{cases}$$

is imposed, while on the outflow

$$\Gamma_{OUT} := \{\boldsymbol{x} \in \overline{\Omega} \ : \ x_1 = 1\}$$

we impose a homogeneous Neumann boundary condition (see Figure 3.2). On the remaining part $\partial\Omega \backslash \{\Gamma_{IN} \cup \Gamma_{OUT}\}$ we consider a Dirichlet boundary condition of zero velocity. According to the inflow boundary condition, the initial velocity is set to $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}_0(\boldsymbol{x}) := \boldsymbol{0}$, $\forall \boldsymbol{x} \in \Omega$. Piecewise linear and continuous basis functions are

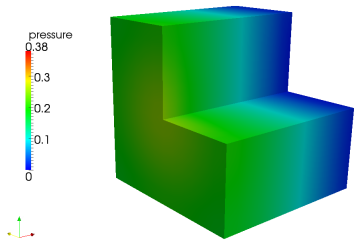
used for both the velocity and the pressure finite element space. We consider the Brezzi-Pitkäranta stabilization [11], given by

$$c_h(p_h, q_h) = \delta \sum_{K \in \tau_h} h_K^2 \int_K \nabla p_h \cdot \nabla q_h dK, \qquad \forall p_h, q_h \in \tilde{W}_p^h(\Omega), \tag{3.13}$$

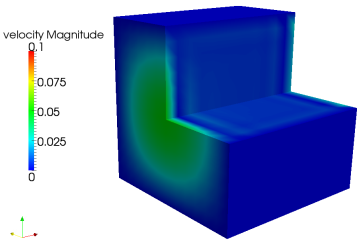
$$G_h(q_h) = 0, \qquad \forall q_h \in \tilde{W}_p^h(\Omega), \tag{3.14}$$

where $\delta > 0$ is a constant, $K \in \tau_h$ denotes a mesh element and h_K its characteristic diameter. One consequence of introducing a stabilization term is that the numerical solution $\boldsymbol{u}_h$ does no longer satisfy the weak form of the incompressibility condition $\nabla \cdot \boldsymbol{u} = 0$. Through the coefficient δ it is possible to minimize the influence of the stabilization term on the divergence of the numerical solution by keeping the problem stable. In the following results it is fixed to $\delta = 10$. For the time integration we use an Euler scheme (BDF1) for $n = 0$, a BDF2 scheme for $n \geq 1$ and a fixed time step of $\Delta t = 5 * 10^{-3} s$. In Figure 3.3 the evolution of the pressure and the magnitude of the velocity field at time steps $50ms$, $100ms$, $150ms$ and $250ms$ are illustrated. During the acceleration period in the time interval $(0, 100ms)$, we observe an increasing pressure near the inflow and some numerical inaccuracies in the velocity field (Figures 3.3a-3.3d). In the following time steps, until $T = 250ms$, we use a constant inflow condition, what results in a stabilized velocity field $\boldsymbol{u}$ and a decreasing pressure p (Figures 3.3e - 3.3h). As mentioned above, the stabilization term influences the incompressibility condition. In Figure 3.4 we report the divergence of the velocity field computed graphically in Paraview at time $t = 250ms$. We observe that the divergence is different from zero mainly near the in- and the outflow faces, what is a consequence of the sinusoidal inflow shape and the homogeneous Neumann condition used on the outflow, respectively. It would be possible to replace the Neumann condition by a boundary condition enforcing zero velocity components in y- and z-direction.

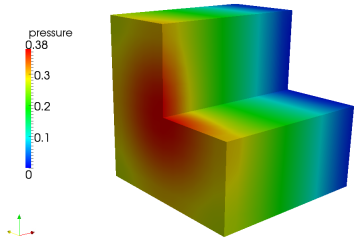
For the next test we keep the settings previously introduces and we investigate the convergence rate of the PCG using Algorithms A, B and C for FETI-DP. The stopping criterion for the PCG is based on the relative reduction of the initial residual by a factor of 10^{-7}. Table 3.1 reports the number if PCG iterations for the first and the second time step by keeping the ratio H/h fixed and increasing the number of subdomains. In the Table 3.1 we can not detect an upper bound for the number of iterations. We notice that for the algorithm using face averages (Alg. C) as additional primal constraints, the second order time integration scheme BDF2 results in a better convergence rate. We observe that in general the number of iterations is the lowest in the case of algorithm C. The increasing number of iterations can be an effect of the stabilization term or of the choice of δ what needs further investigations.



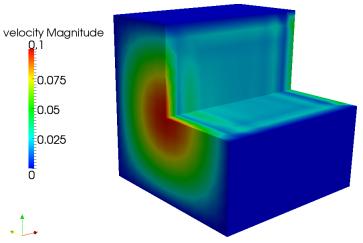
(A) Pressure at $t = 0.05s$



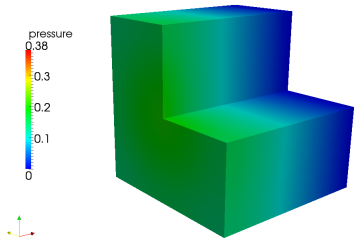
(B) Velocity magnitude at $t = 0.05s$



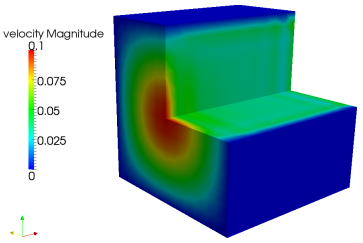
(C) Pressure at $t = 0.1s$



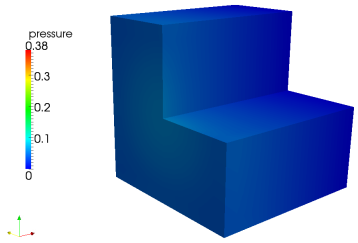
(D) Velocity magnitude at $t = 0.1s$



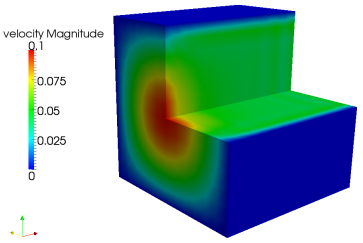
(E) Pressure at $t = 0.15s$



(F) Velocity magnitude at $t = 0.15s$



(G) Pressure at $t = 0.25s$



(H) Velocity magnitude at $t = 0.25s$

FIGURE 3.3: Evolution of the pressure and the magnitude of the velocity field at time steps $50ms$, $100ms$, $150ms$ and $250ms$ in a cut of the domain.

3.5 Implementation aspects of FETI-DP

The aim of this section is to provide a brief overview of the main implementation aspects of the code developed for FETI-DP, as well as for the Finite Element approximation of the Poisson and Navier-Stokes problems. We used Matlab version R2012b with student

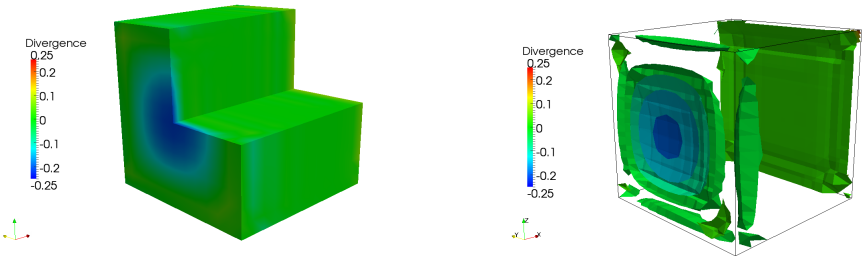


FIGURE 3.4: The divergence of the velocity field inside the domain (left) and its non-zero contours (right) at time $t = 250ms$.

H	N_s	H/h	Dofs/Subd	Dofs	N_{Iter} $(n = 0/n = 1)$		
					Alg. A	Alg. B	Alg. C
1/2	8	4	500	3979	18/19	16/18	13/14
1/3	27	4	500	13'332	30/31	26/27	18/19
1/4	64	4	500	31'433	44/44	34/34	24/22
1/5	125	4	500	61'156	55/57	45/46	35/32
1/6	216	4	500	105'375	71/68	60/57	44/41

TABLE 3.1: Number of PCG iterations using a FETI-DP with stabilization term (3.13).

license. The code used for the numerical simulations has been entirely developed in the framework of this master project; namely both the Finite Element and the FETI-DP parts. Furthermore, we highlight that no additional third party library has been used. In the following we summarize the main different parts of the code developed; to this end, in Figure 3.5 a block diagram is presented, illustrating the principal functionalities of the code infrastructure. Henceforth, we summarize the key Matlab functions implemented to deal with Navier-Stokes problem using FETI-DP:

- `solveNavierStokes_3D` is the main function for solving numerically an incompressible Navier-Stokes problem in a cube. It performs preprocessing such as the mesh generation (`generateMesh_3D`), decomposition of the domain into $N_s = N \times N \times N$ subdomains (`generateSpaceDec_3D`) and the finite dimensional representation of the lifting function (`generateDirichletBndCond_3D`).

- `FETI_DP_solve_NS_3D` contains two main parts of the FETI-DP solving procedure for a time dependent problem. First, we perform problem specific preprocessing for assembling the local matrices (`generateSubAssemblingABC_NS_3D`), the time independent part of the forcing term (`generateSubAssemblingF_NS_3D`), the coarse matrix S_Π (`generateCoarseMatrix_NS_3D`), the local continuity operators $B_\Delta^{(s)}$ (`generateContOpB_3D`), the local matrices of additional constraints $Q_\Delta^{(s)}$ (`generateAddOpQ_3D`) and we precompute the local mixed problems ${K_{RR}^{(s)}}^{-1} K_{R\Pi}^{(s)}$

(`performPreCompPrimal_NS_3D`). In the second part, the function runs the time dependent part of the problem, namely at each time step we update the right hand side (`generateSubAssemblingRHSNonLin_NS_3D` and `generateSubAssemblingRHSTimeDisc_NS_3D`)

- `FETI_DP_solveDualProb_NS_3D` uses the preprocessed data to set up and solve the preconditioned dual problem by the PCG method. The two main tasks performed during the solution of the linear system are the application of the Dirichlet preconditioner (`FETI_DP_applyDirPrecond_NS_3D`) and the application of the dual matrix (`FETI_DP_applyDualMat_NS_3D`).

All the matrices and maps that are repeated for the three vectorial components of the velocity field, are stored only once to save memory. In the application of the FETI-DP method to a problem, we need to identify the local indices of the internal, dual and primal dofs in each subdomain. Regarding this assignments, there are two main possibilities:

1. Store the global assignments of the nodes, and for each subdomain use a global to local map in order to compute the local assignments.
2. For each subdomain store the local assignments of all the local dofs.

The first one (used in our implementation) uses less memory but is also less efficient because the local assignments have to be recomputed several times in different functions.

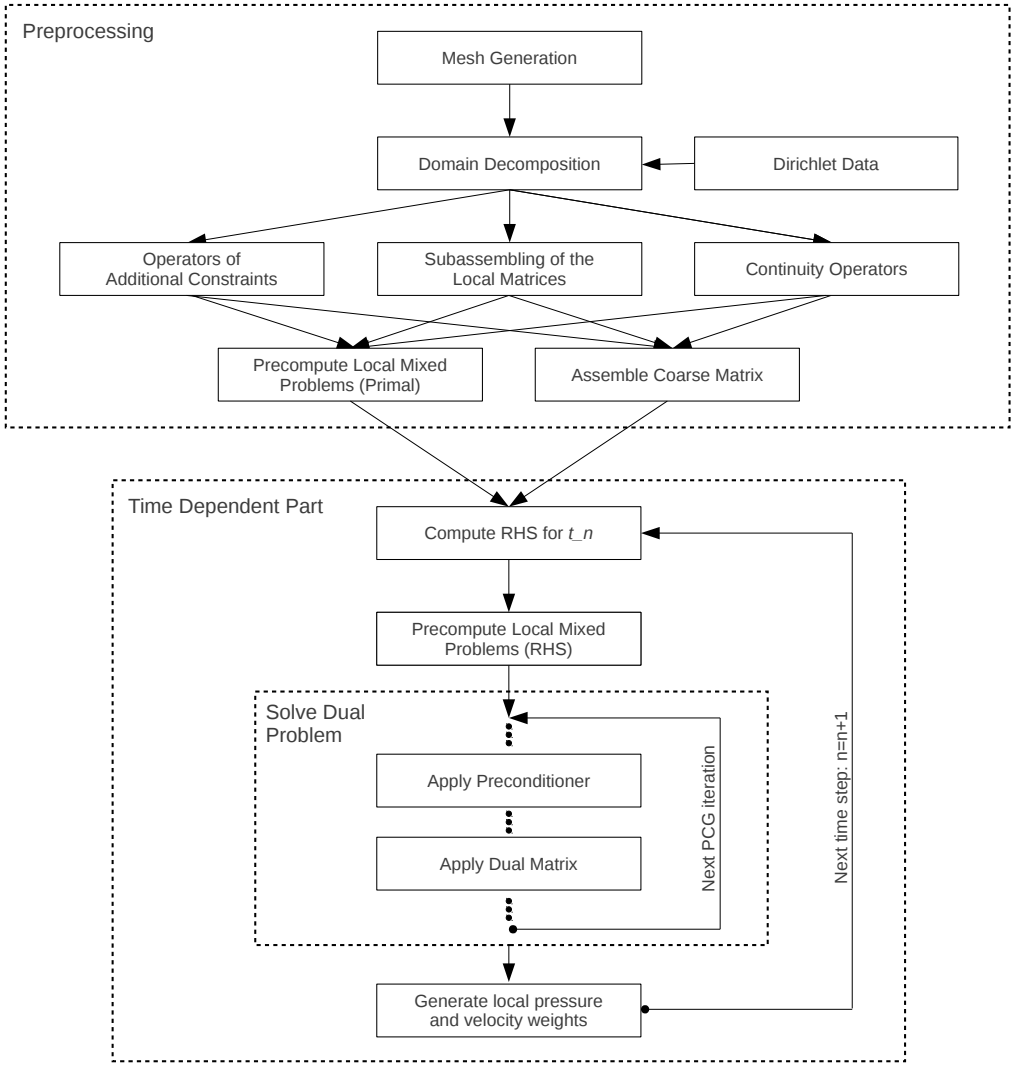


FIGURE 3.5: Block diagram illustrating the strategy of solving an incompressible Navier-Stokes problem in a 3-dimensional domain.

Chapter 4

Conclusions and Perspectives

In this work we have considered the FETI-DP domain decomposition technique applied to the Poisson and Navier-Stokes problems discretized by means of the Finite Element method. The highlight of this thesis is mainly focused on the theoretical analysis and algorithmic aspects of FETI-DP and on its application to the Poisson and the Navier-Stokes problems.

In the first part of this work we have recalled the FETI-DP method and applied it to the Poisson problem. In particular we have investigated the preprocessing effort required on the subdomain level to identify the internal, dual and primal degrees of freedom. In order to minimize the computational work in the preprocessing, it may be convenient to deal with structured subdomains.

FETI-DP is a dual iterative substructuring method wherein the iterates (parallel between the subdomains) are discontinuous across the subdomain interfaces; these jumps disappear only at convergence. We have considered different algorithms that allows for better convergence results, namely adding extra primal constraints based on edge/-face averages. When dealing with 2-dimensional problems it is sufficient to use primal constraints at the vertices of the subdomains, while for 3-dimensional problems the additional average constraints on the subdomain edges or faces turn out to improve significantly the convergence rate of the PCG method.

In the last part of this thesis we have used FETI-DP to deal with the fully explicit Navier-Stokes equations in a three dimensional setting. We have considered piecewise linear basis functions for both the pressure and the velocity fields, combined with the Brezzi-Pitkaranta pressure laplacian stabilization. At the basis of the numerical results presented in this chapter we have noticed that FETI-DP applied to such a problem does not provide scalable results in terms of iterations for the solution of the dual problem.

A possible extension of this work is represented by the study the properties of FETI-DP method applied to the Navier-Stokes equation using either stable pairs for basis functions for the pressure and the velocity, or using piecewise linear bases combined with a different stabilization technique such as the Streamline Upwind Petrov-Galerkin technique to further study their effect on convergence rate of the dual problem. Finally, from the implementation point of view, it would be worthwhile to port the serial Matlab code fully developed in this work to a parallel framework in order to get significant improvements of the performances.

Appendix A

Saddle-Point Problems

We briefly recall the conditions given in [10] under which a variational problem is equivalent to a saddle-point problem. The saddle-point problem corresponds to a constrained variational problem being general enough to include the problems formulated by the Galerkin method treated in the previous chapters.

Problem Formulation

Let $(X, \langle\cdot,\cdot\rangle_X)$ and $(M, \langle\cdot,\cdot\rangle_M)$ be two Hilbert spaces with corresponding norms $||\cdot||_X$ and $||\cdot||_H$, respectively. We introduce two bilinear forms $a : X \times X \to \mathbb{R}$ and $b : X \times M \to \mathbb{R}$ that we suppose to be continuous, i.e. there exist two constants $C_a > 0$ and $C_b > 0$ such that:

$$|a(u,v)| \leq C_a||u||_X||v||_X \quad \text{and} \quad |b(u,\lambda)| \leq C_b||u||_X||\lambda||_M, \tag{A.1}$$

for all $u, v \in X$ and $\lambda \in M$. Let X' and M' denote the dual space of X and M, respectively; for $f \in X'$ and $\sigma \in M'$ we consider the following constrained problem:

$$\text{find } (u,\lambda) \in X \times M \quad : \quad \begin{cases} a(u,v) + b(v,\lambda) = \langle f, v\rangle & \forall v \in X \\ b(u,\mu) = \langle \sigma, \mu\rangle & \forall \mu \in M \end{cases} \tag{A.2}$$

where $\langle\cdot,\cdot\rangle$ denotes the duality between a Hilbert space and it's dual space. We associate the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ with the linear operators $A : X \to X'$ and $B : X \to M'$ defined by the relations:

$$\begin{aligned} \langle Au, v\rangle &= a(u,v) \qquad \forall u, v \in X, \\ \langle Bu, \lambda\rangle &= b(u,\lambda) \qquad \forall u \in X, \lambda \in M. \end{aligned}$$

We recall that for a linear operator $F : V \to W$ between two Hilbert spaces U and V, the adjoint operator $F^{\mathrm{T}} : W' \to V'$ is defined by:

$$\langle F^{\mathrm{T}}\varphi, v\rangle = \langle \varphi, Fv\rangle, \qquad \forall v \in V, \varphi \in W'$$

Moreover, for the operator B associated to the bilinear form $b(\cdot,\cdot)$, the adjoint operator $B^{\mathrm{T}} : M \to X'$ is defined by:

$$\langle B^{\mathrm{T}}\lambda, u\rangle = \langle Bu, \lambda\rangle = b(u,\lambda), \qquad \forall u \in X, \lambda \in M.$$

Hence, the constrained problem (A.2) can be restated as the following *saddle-point problem*:

$$\text{find } (u,\lambda) \in X \times M \quad : \quad \begin{cases} Au + B^{\mathrm{T}}\lambda = f & \text{in } X', \\ Bu = \sigma & \text{in } M'. \end{cases} \tag{A.3}$$

Problem Analysis

To analyse the saddle-point problem (A.3) we introduce the affine manifold

$$X^{\sigma} := \{v \in X \ : \ Bv = \sigma \text{ in } M'\}$$

and the subspace

$$X^0 := \{v \in X \ : \ Bv = 0 \text{ in } M'\} = \ker(B).$$

Since the operator B is continuous, the subspace $X^0 = B^{-1}\{0\}$ is closed and therefore a Hilbert space. We can then associate (A.2) with the reduced variational problem:

$$\text{find } u \in X^{\sigma} \quad : \quad a(u,v) = \langle f, v\rangle \qquad \forall v \in X^0. \tag{A.4}$$

If $(u,\lambda) \in X \times M$ is a solution of (A.2), then it is straightforward to see that u is a solution of (A.4). The following theorem given in [10] introduces suitable conditions that allow the converse to hold too.

Theorem A.1. *Let the bilinear form $a(\cdot,\cdot)$ satisfy the continuity condition (A.1) and be coercive in the space X^0, that is*

$$\exists \alpha > 0 \ : \ a(u,u) \geq \alpha ||u||_X^2 \quad \forall u \in X^0.$$

Suppose moreover that the bilinear form $b(\cdot,\cdot)$ *satisfies the continuity condition (A.1) and the compatibility condition:*

$$\exists \beta^\star > 0 \ s.t. \ \forall \lambda \in M, \exists u \in X, \ with \ x \neq 0 \ : \ b(u,\lambda) \geq \beta^\star ||u||_X ||\lambda||_M.$$

Then for every $f \in X'$ *and* $\sigma \in M'$*, there exists a unique solution* $u \in X^\sigma$ *of problem (A.4); furthermore, there exists a unique* $\lambda \in M$ *s.t.* (u,λ) *is the unique solution to the original constrained problem (A.2).*

Bibliography

[1] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method–part I: A faster alternative to the two-level FETI method. *International Journal for Numerical Methods In Engineering*, (50):1523–1544, 2001.

[2] J. Gallier. The schur complement and symmetric positive semidefinite (and definite) matrices. 2010. University of Pennsylvania.

[3] P. Gervasio, F. Saleri, and V. Veneziani. Algebraic fractional-step schemes with spectral methods for the incompressible Navier-Stokes equation. *Journal of Computational Physics*, (214):347–365, 2006.

[4] D. Keyes, Y. Saad, and D. Truhlar. *Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engeneering.* Society for industrial and applied mathematics, USA, 1995.

[5] A. Klawonn and O. Widlund. Dual-primal feti methods for linear elasticity. *Communications on Pure and Applied Mathematics*, (LIX):1523–1572, 2006.

[6] G. Leoni. *A First Course in Sobolev Spaces.* American Mathematical Society, United States of America, 2009.

[7] J. Li. *Dual-Primal FETI Methods for Statinary Stokes and Navier-Stokes Equations.* PhD thesis, New York University, 2002.

[8] J. Nocedal and S. Wright. *Numerical Optimization.* Springer Verlag, Berlin Heidelberg, 2006.

[9] N. Ottosen and H. Petersson. *Introduction to the Finite Element Method.* Prentice Hall International (UK) Ltd, Hertfordshire, 1992.

[10] A. Quarteroni. *Numerical Models for Differential Problems.* Springer Modeling, Simulation and Applications, Berlin Heidelberg, 2014. 2nd ed.

[11] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer Series in Computational Mathematics, Springer-Verlag Berlin Heidelberg, 1994.

[12] A. Quarteroni, J. Periaux, Y. Kuznetsov, and O Widlund. *Domain Decomposition Methods in Science and Engineering*. American Mathematical Society, USA, 1992.

[13] O. Rheinbach. *Parallel Scalable Iterative Substructuring: Robust Exact and Inexact FETI-DP Methods with Applications to Elasticity*. PhD thesis, Universität Duisburg-Essen, 2006.

[14] D. Rixen and C. Farhat. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods In Engineering*, (44):489–516, 1999.

[15] A. Toselli and O. Widlund. *Domain Decomposition Methods - Algorithms and Theory*. Springer Series in Computational Mathematics, Berlin Heidelberg, 2005.